

Copyright is owned by the Author of the thesis. Permission is given for a copy to be downloaded by an individual for the purpose of research and private study only. The thesis may not be reproduced elsewhere without the permission of the Author.

Development of Flexible Education Systems for Technology Students

A thesis presented in partial
fulfilment of the requirements
for the degree of
Master of Technology
in Computer Systems Engineering
at Massey University

Vivienne Breen

1998

Abstract

The pressures on educational institutions to keep pace with the changes in educational theory and technology are growing rapidly. There is now more competition between education providers, especially in the tertiary and career training sectors, this has meant that students and industry is demanding more input into the education process. As a result a more flexible approach is being taken to the delivery of courses. This thesis describes the development and implementation of a flexible learning approach applied to technology related subjects. It addresses the work carried out in relation to a specific aspect of the Bachelor of Technology degree as taught at Massey University, Palmerston North, New Zealand. Issues have been addressed in educational programme design, material presentation, and a major focus for implementation has been computer mediated assessment mechanisms. A guide has been developed to assist the educator in increasingly applying flexibility to subjects undertaken within the Bachelor of Technology degree reflecting the specific needs of the New Zealand industrial and educational sectors.

Key words: Flexible education, flexible learning, computer mediated education, computer assisted learning, computer simulation is assessment, computerised marking, computer mediated assessment.

Acknowledgements

There are many people who have contributed to the work undertaken within this thesis and would be too numerable to mention each by name. I wish to mention specifically the support of the Department of Production Technology at Massey University and the assistance and patience of my supervisor Professor D J Barnes. My thanks to you all for your assistance and support throughout.

“Learning is more effective when it’s fun”

Peter Klein

Table of Contents

DEVELOPMENT OF FLEXIBLE EDUCATION SYSTEMS FOR TECHNOLOGY STUDENTS.....	1
ABSTRACT	2
ACKNOWLEDGEMENTS	2
TABLE OF CONTENTS	3
FIGURES AND TABLES.....	5
1 INTRODUCTION AND OVERVIEW	6
1.1 INTRODUCTION	6
1.2 OUTLINE OF THE CHAPTERS	7
2 THE CHANGES IN TRADITIONAL EDUCATION AND TRAINING	9
2.1 THE WAY WE DO THINGS.....	9
2.2 ATTITUDES AND APPROACHES	10
2.3 APPLICATION OF STANDARD THEORY IN NEW WAYS.....	13
2.4 STRUCTURE OF PROGRAMMES, POTENTIAL USES.	15
2.5 BARRIERS TO CHANGE	17
2.6 EVALUATING THE TEACHING AND THE LEARNING.....	18
2.7 THE EDUCATION OF TECHNOLOGISTS AND ENGINEERS	19
2.8 THE IMPACT OF TECHNOLOGY ON WHAT IS TAUGHT	23
2.9 CONCLUDING COMMENTS.....	24
2.10 REFERENCES.....	25
3 REVIEW OF EXISTING TECHNOLOGICAL SYSTEMS AND SYSTEM COMPONENTS FOR FLEXIBLE EDUCATION	30
3.1 INTERACTION AND INTERFACES	30
3.2 AUTHORING ENVIRONMENTS AND TOOLS	32
3.3 MULTIMEDIA MATERIALS	34
3.4 HYPERMEDIA ENVIRONMENTS	37
3.5 VIRTUAL REALITY AND SIMULATION	41
3.6 HARDWARE, COMMUNICATIONS AND SUPPORT.....	43
3.7 CONCLUDING COMMENTS.....	48
3.8 REFERENCES.....	50
4 PROGRAMME DEVELOPMENT.....	53
4.1 COURSE DESIGN.....	54
4.2 TOPIC SELECTION.....	55
4.3 DELIVERY MECHANISMS	57
4.3.1 <i>Asynchronous delivery</i>	57
4.3.2 <i>Synchronous delivery</i>	59
4.4 STUDENT ASSESSMENT	59
4.4.1 <i>Tutorial examples</i>	60
4.4.2 <i>Self assessment modules</i>	60
4.4.3 <i>Assignments</i>	61
4.4.4 <i>Formal examinations</i>	61
4.5 DATA SECURITY.....	61
4.6 COURSE EVALUATION.....	62
5 PROGRAMME IMPLEMENTATION, EVALUATION AND REFINEMENT	63
5.1 INFORMATION DELIVERY	63
5.1.1 <i>Instructions</i>	64
5.1.2 <i>Interactive learning</i>	71
5.1.3 <i>Integration of tutorial aspects into course material</i>	76
5.1.4 <i>Additional help</i>	86

5.2 ASSESSMENT MECHANISMS	87
5.2.1 Assignments.....	88
5.2.1.1 Statistical Quality Control Assignment - Year One.....	89
5.2.1.2 Experimental Design and Multivariate Analysis Assignment - Year One.....	92
5.2.1.3 Basic Statistics and Design of Experiments Assignment - Year Two	94
5.2.1.4 Experimental Design and Multivariate Analysis Assignment - Year Three	95
5.2.2 Formal examinations	97
5.3 MACRO DESIGN AND CONSTRUCTION	97
5.4 CONCLUSIONS	99
5.4 REFERENCES.....	102
6 IMPACT OF EXPERIMENTAL WORK ON THE TEACHING PROGRAMME.....	103
6.1 INTRODUCTION AND EFFECTS OF CHANGES MADE.....	103
6.2 EFFECTS ON THE ASSISTANCE AND EXPLANATION REQUIRED	105
6.3 GUIDELINES FOR FUTURE IMPLEMENTATION OF CHANGE IN THE ASSESSMENT PROCESS	106
6.4 OUTCOMES OF THE EXPERIMENTAL WORK.....	107
6.5 FUTURE IMPLEMENTATIONS.....	110
6.6 REFERENCES.....	112
APPENDIX A COURSEWARE DEVELOPMENT TOOLS - EVALUATION CHECKLIST SUGGESTION.....	113
APPENDIX B AUTHORIZING LANGUAGES AND ENVIRONMENTS	115
APPENDIX C ASSIGNMENT 1 YEAR 1.....	126
C.1 ASSIGNMENT QUESTION SHEET	126
C.2 MACRO FILES.....	127
APPENDIX D EXPERIMENTAL DESIGN ASSIGNMENT - YEAR ONE	137
D.1 ASSIGNMENT QUESTION SHEET	137
D.2 MACRO FILES	139
APPENDIX E EXPERIMENTAL DESIGN ASSIGNMENT - YEAR TWO	149
E.1 ASSIGNMENT QUESTION SHEET.....	149
E.2 MACRO FILES.....	150
APPENDIX F DESIGN OF EXPERIMENTS AND MULTIVARIATE ANALYSIS ASSIGNMENT - YEAR THREE.....	171
F.1 ASSIGNMENT SHEET	171
F.2 MACRO FILES	174

Figures and Tables

Figure 2.1	Page 22
Figure 3.1	Page 31
Figure 5.1	Page 66
Figure 5.2	Page 67
Figure 5.3	Page 68
Figure 5.4	Page 69
Figure 5.5	Page 70
Figure 5.6	Page 74
Figure 5.7	Page 75
Figure 5.8	Page 77
Figure 5.9	Page 78
Figure 5.10	Page 79
Figure 5.11	Page 80
Figure 5.12	Page 81
Figure 5.13	Page 82
Figure 5.14	Page 83
Figure 5.15	Page 84
Figure 5.16	Page 85
Figure 5.17	Page 86
Table 4.1	Page 57
Table 5.1	Page 91
Table 5.2	Page 100

Trademarks are the property of their respective Companies and are referenced at the first occurrence within the text.

1 Introduction and Overview

1.1 Introduction

This thesis addresses the issue of flexible education as applied to technology related subjects. Specific attention is given to the development of a flexible course structure and the implementation of assessment units.

In the face of a changing educational environment a more flexible approach is increasingly being adopted. Flexible learning seeks to enable students to learn at their own pace, in a preferred location, the topics of interest within a given qualification structure, and at a time more convenient to themselves and in a manner that maximises their learning. The provision of such programmes frees both the student and educational institution to focus on the material being taught, and to assist students to understand and implement what they are learning.

The introduction of flexible programmes of study is being enabled in part by the development of computer and communication technologies. No longer is distance a limiting factor in the delivery of education. With the use of conferencing technologies the interaction between students, and between student and tutor is maintained independent of location. Of special significance is the developments in computer technologies, both in terms of available hardware and software. It is increasingly possible to view live video images of people and machinery in real time. The computer control of such machinery has enabled the complete remote operation. Thus enabling a student, or worker, to control the actions of a machine, be it an experimental apparatus or a production model, from any location and see the response to commands issued with a minimal delay.

Where a computer forms the central delivery or communication passing agent in a flexible learning environment this can be termed Computer Mediated Education. The ability of a computer to perform repeated tasks and its relation to the drill and practice exercises of many existing educational systems has long been established. A number of systems are available that use this method to instruct the user. Some subjects relate specifically to this type of learning and they should continue to be used. The subject areas addressed in this thesis are those that primarily do not fall into this category. A matching of subject matter and ways available to assist in its teaching was made, and where possible implemented. Some of the possible solutions fell well beyond the limitations of time and financial availability for both the work contained in this thesis and that of most educational institutions. It is acknowledged that the reality of life is that the available level of finance and other resources is not infinite. A perfect solution may be implemented with unlimited resources, yet here we have sought to work within strict limitations to establish a good and valid solution that is both sustainable and adoptable.

Amongst the increasing pace of change in technologies available has been changes in the focus and requirements of education. The requirements of tertiary level education especially as it relates to professional training, for example doctors, engineers, veterinarians, technologists, is constantly on the move as the professions themselves adapt the changes in society and technological developments. Tertiary institutions are under pressure to keep up with these changes, with the pressure coming from the industry for which students are being prepared and from the

students wishing to equip themselves for future careers. Flexibility within a course of study is one way to address both of these issues. It enables the establishment of modularised subjects which can be updated faster and with greater accuracy than before, and to establish greater links with those in industry resulting in a shorter gap between the subject studied at university and that which is later applied.

The rate of change now being experienced is reflected in the need to acquire life long learning skills and attitudes. No longer is it possible to remain in the same job or forty years without the job changing dramatically. Graduates now face the prospect of re-education and re-training at least three to four times in their working lives. The implementation of a flexible learning programme enables the student to discover and use the ways in which they learn best. The knowledge of how they learn best, and what support activities are of benefit, is applicable to any training or education undertaken. Education is coming to be viewed as a partnership between the institution represented by its various tutors and the student. No programme is fully successful unless both parties actively engage in the learning exercise to establish, measure and achieve the desired goals.

The joint meeting of educational goals and the utilisation of appropriate technologies has been a major focus of the work carried out within this thesis. The thesis itself presents both sides in order to demonstrate that the purposes of education, the needs of industry for trained graduates, and the preparation of students for life long learning, as especially applied to technology subjects, can be achieved.

1.2 Outline of the chapters

Chapters 2 and 3 focus on existing research and system implementations. Chapter 2 from an educational perspective and Chapter 3 from a technological and systems perspective. Current practices and why they are in place is discussed in Chapter 2, along with proposes ways to implement new systems form the existing educational theories. The resistance to any change is presented along with issues specific to the education of technologists and engineers, and the impact of changes in available technologies is having upon the their professional career development. Chapter 3 addresses the available technologies and the systems within which they operate in a more detailed manner. The focus changes in this chapter to reflect a more technology orientation and thus more technical detail is presented than that found in Chapter 2. Various mechanisms and environments useful within an educational setting are discussed along with the technical issues of delivering the material in the various formats to a distributed audience.

Chapters 4 and 5 deal with the development of strategies for flexible learning and their implementation as it relates to material presentation and assessment of learning. Chapter 4 outlines the development of a flexible learning approach in relation to the Bachelor of Technology degree and the subjects undertaken within it. It discusses aspects of course design relevant to any subject and the selection of appropriate presentation mechanisms. How the material is to be delivered and presented has its' own effects upon the choices made and the number of different implementations required to accommodate the various learning styles of students. The way a course is assessed is determined in the most part by what mechanisms of delivery and presentation have been used. Options available and the related security factors are also discussed. Chapter 5 presents the implementation of several experimental

systems, two relating to material presentation, and four relating to the assessment of learning. The methods used or material presentation is dependent upon the needs of the audience and the desired use of the material. Two of the available methods are presented in relation to computer based systems for material delivery, they are instructions and interactive learning. As stated earlier assessment units were a specific focus for the experimental work carried out within the scope of this thesis. There is a large section in this chapter presenting the development over time of assignments that have been implemented in a computer mediated fashion. From each implementation improvements were noted and incorporated into the next implementation. As the assessment of assignment work has become increasingly available by these mechanisms so attention is drawn to the more formal aspects of the evaluation of student learning, the final examination. To conclude this chapter the development of the simulation systems behind the assignments is presented with the full programme details contained in the associated appendices. Formal conclusions can be made at this point as to the effectiveness of such implementations which lead to the identification of good systems components and guidelines for further implementations.

Chapter 6 outlines the impact of the experimental work carried out to date, and the implications for further development of a flexible learning approach to technology related subjects, and beyond. It reflects the lessons learnt over the entire period of experimental work and the implications these have for increasing the levels of flexibility within the Bachelor of Technology degree.

2 The Changes in Traditional Education and Training

Education faces the challenge of preparing students to be effective within the society and culture in which they operate. This has been the goal for many years, and educationalists will say it still is. Yet there is a growing disparity between the goal and what is actually happening. Society has changed markedly since Victorian times, but our educational systems in many cases have not. This chapter looks at some of the issues involved, primarily at university level education and the changes occurring within it. Some of these are the breakdown of barriers between university and vocational training, the increasing requirement for flexibility, and the availability of technologies to support educational goals. Some specific examples of various systems, their implementation and evaluation are discussed in Chapter 2.

2.1 The way we do things

Education today, especially the education and re-education or retraining, of adults must by necessity take into account the development in technology and educational theories. "In the last twenty years or so, a number of factors have contributed to a major change in the status of distance education as an appropriate and effective mechanism for education and training of adults. Economic changes have radically altered the skills required by enterprises. This has meant rethinking the objectives of, in particular, the initial education system. The pace of economic change has also led to an increased demand for continuing education provision. The large number of people made redundant from declining traditional industries has also produced a need to retrain the unemployed. Significant changes in demographic profiles that are under way or predicted, have also prompted a renewed concern to ensure there will be a sufficient supply of necessary key skills for future economic demands. Advances in new educational technologies have further stimulated the interest in distance education, as an integral part of existing educational strategies" [1].

The changing demands of education not only affect distance learning but also the more common internal or full-time course. There is an increasing need for more flexibility in the programmes offered and the teaching methods used. The student or learner is required to take a more active part in the process. Particularly in vocational or professional areas where the range of material required to be learnt by students to equip them for their future careers has increased considerably. New technologies and the increased participation in tertiary education in almost every country has increased the material to be covered or alternatively the options which need to be offered within existing degree courses. [2,45] The emphasis on experiential learning relevant to the students future careers also places considerable pressure to expand curricula to include a wider range of subject material than that required previously. All of this places pressure on the facilities of an institution, increasing the teaching loads of staff and the work loads of students. To meet these changes more flexibility is required in tertiary education programmes.

For the educational provider or institution the advances in technology and in the understanding of how people learn creates ongoing challenges for continuous

improvement. One aspect that is easily observed in the rise in student numbers able to be handled at one time. In 1974 it was written that “advances in the technology of education, despite dangers, is technology and the implications of technological advances is essential if higher education is to deal effectively with reality” [3]. Although many years have passed since then these statements are no less relevant, probably more so.

With these increasing requirements and pressures on continuous professional development, both the context in which material is learnt and the way in which levels of competence in the application of this material are assessed, are under constant review and revision [4,5,6]. The matching of the learning and assessment methods is alleviating some the stresses involved in both the assessment itself and the later implementation of acquired skills in the workplace [10,6,4].

The traditional methods of presenting material at University level has been the lecture. This, generally, is a passive activity which does not require the students to actually participate except where the lecturer sets up specific interaction with the class or between members of the class in the form of discussion, group activities or similar activities. These later interaction methods represent trends in increasing the effectiveness of the lecture approach. Group interaction, tutorials and computer assisted learning form growing areas for increasing the effectiveness of modern tertiary education. Computers assisted learning has been used extensively in vocational training [3]. It has not had the same impact in formal tertiary education. There is, however, a growing awareness of the potential of computer assisted learning in the tertiary environment [7].

The importance of good design of the entire educational course or programme of study is often reflected in the matching of educational goals and the teaching/learning methods used. This means trying to ensure that the course and the students ‘fit together’, by the use of a variety of presentation and learning methods, and that the course and assessment that is used also ‘fit together’. That is not just teaching to the test, but developing teaching and assessment valid for the material and purposes of instruction [8,9]. Evaluation and assessment in any course of study measures both what the student has learnt and the effectiveness of the course itself and its constant improvement [10].

2.2 Attitudes and approaches

The changes in industry and society in general are leading educators to rethink what is to be taught and the methods by which this should be done. The meanings, processes and mechanisms used in education are under a microscope. Although this revision is a good thing not everyone can, or probably ever will, agree on what is ‘good’ and ‘fit’, what is ‘appropriate’ and how to assess or measure it.

Changes in attitude within educational institutions is often reflected in the policies they set and enact, both at institutional and departmental level. A movement towards increased use of computers as tools for information processing, as well as teaching aids is occurring in most places of ‘higher learning’. It is in fact happening throughout the whole educational sphere. Although computers, with the associated

capabilities of multimedia and internet communication, have and are the main focus they should not be seen in isolation. Staff development must move hand in hand with technological development to ensure effectiveness in the educational programmes [11, 12]. Institutions themselves must also ensure that adequate provision is made in terms of learning spaces (ie rooms), equipment and relevant expertise. “Less than two generations ago the opportunities for education were sufficient for the demand. There was little talk in those days about shortage of teachers or lack of accommodation. Therefore there was no need to examine critically the technology of education. All this has changed. In many countries there is a grave shortage of teachers, especially in universities and colleges of technology, and the demand for post-secondary education outruns finance, accommodation and equipment. In some places it assumes the dimensions of an intellectual famine. It is not surprising that nations are beginning to pay attention to the technology of education and to ask where the application of scientific analysis to educational systems is likely to be rewarding, and where it could be irrelevant or damaging” [3].

The emphasis within education is changing from ‘you must know this’ to ‘preparation for the future’. What educators think of as the purpose of learning has a great influence upon not just what is taught but how it is taught [13]. A good place to observe this shift is in the tertiary education arena. The guiding philosophy behind some programmes of study at university level is changing, enabling it to encompass aspects of the philosophies of vocational training. In the past a greater distinction has been maintained between trade or vocational training and university studies in general. However, careers such as medicine, law and veterinary science have also been an element of university programmes for many years. Part of the shift to the recognition of vocational aspects of university education has been caused by the changing nature of industry and its employment requirements. A number of ‘traditional’ jobs for both trade and university graduates have disappeared, or have been redesigned. This is due in the main part to advances in technology and ‘primary’ level education [14]. “Three major environmental forces are pushing higher education towards vocationalism while the internal heredity of universities retains its non-vocational aim. Mass systems of higher education can offer excellence in both vocational and non-vocational programmes if universities do not permit their non-vocational programmes to be used by employers as screening devices for jobs, if they maintain opportunities for intellectual talent and avoid the inappropriate use of cost-benefit analysis, and if they develop mission-oriented studies as well as discipline-oriented programmes” [15].

Technological advancement and the changes in the structure of companies and how they operate, is reflected in their demands for skills and knowledge from employees. This demand is passed back to the educational institutions, who are working increasingly with industry to match what is taught to what is needed. The increased interaction between higher education and industry is not just aimed at the production of a competent and sustainable workforce, but is fuelling the fires of change within the educational institutions themselves [6, 5]. Existing learning methods that are still being used in educational institutions include the lecture, tutorial, set readings and written examinations. There is a trend towards a more flexible and mobile learning environment [16]. This has resulted in the greater emergence of computer assisted learning in to the tertiary education environment. Educators in seeking to cope with the changing needs of employers are becoming increasingly aware of a need to match

the material and methods of learning in an appropriate manner. To minimise the limitations of each method, normally the passivity of the lecture and readings style of learning, and to take full advantage of the greater opportunities the differing methods allow for increased student responsibility for their own learning.

The fact of life in today's educational fields is that individual institutions are fighting for their own survival. With increasing demand on diminishing resources educators are being faced with a decision to 'change or die' [11, 14]. The use of technology, and especially computers, is enabling the provision of a greater number of educational programmes to a wider audience. Today it is not just the use of a computer to duplicate the rote learning of the past, but coupled with multimedia capabilities the development of entirely new worlds are possible, where you learn by being a part of this environment, by interacting with it. As we become increasingly accustomed to these types of interactions, mainly through computer/video games, the developers (and educators) of such environments need not only to capture the 'knowledge factor' but also the 'entertainment factor' [17]. As most of us know learning is easier and we learn more when we are having fun [44]. "Any technology which increases the rate of learning enables the teacher to teach less and the learner to learn more" [3], and learning is the true goal of education.

When we move towards a greater use of computer assisted learning environments consideration must be given, not only to the construction of the system itself, but also to the attitudes and introduction of the students to using each learning tool. Students can be classed as 'computerphobic' or more generally 'computer shy', where they have a higher level of distrust of the machine. They do not feel comfortable relying on a machine, but would rather 'prove' it or 'do' it themselves, either in conjunction with the computer system or instead of it. One of the many studies into the issues of attitude and gender to the use of computer based educational systems concludes that, "students had a positive attitude towards using CAL but were not prepared to rely on this learning method. Overall no significant gender differences were found, except in a preference for use, males more" [18]. This shows that with the right introduction, familiarisation and encouragement there are no real barriers to using computer based learning tools for effective learning. Other students may even prefer the computer based studies in preference to existing methods. In many cases a student is content to learn from a computer based learning environment, the material presentation is well constructed and encourages interactive responses, but the associated help or documentation is more of a hindrance to the effective use of the systems and so is not used [19]. These two points are covered by Duin [20] as follows, "effective computer programs make learners active participants instead of passive absorbers of the content being studied. Effective computer documentation should also allow learners to create, explore, and integrate knowledge as they use the computer system. While computer programs as a whole are improving, the documentation that accompanies the programs often lags behind in terms of centering on learners needs".

The increasing use of multimedia and computer based instructional packages does not eliminate the tutor or educator, it merely redefines their role. Provision for the development of their skills in interacting with students and these new educational tools should not be neglected [13]. Even from a developer's point of view, "whether multimedia instruction replaces or works with a human instructor, most developers agree that the best way to make it appealing is to duplicate many of the best aspects of

a personal teacher” [17]. Even within the traditional structures of university education, particularly the lecture environment, changing attitudes to the use of technology can be seen in the differing methods of material presentation. No longer is it a case of ‘you listen while I talk’, but more of enabling students to utilise how they best learn by providing the material in more than one format [21, 22]. While wishing to provide the best environment in which their students can learn tutors must also guard themselves against the traps that can ensure that students are overwhelmed by the presentation rather than making it easier for them to understand the material. Other challenges facing the tutor include the need for constant retraining, a natural resistance to change, the danger of change for its own sake, and losing sight of the relationship with educational theory. Tutors should always ask themselves two questions in regard to any presentation, method of interaction or assessment to be undertaken, (1) why is it done, to what purpose? and (2) is it doing anything in terms of student learning?

2.3 Application of standard theory in new ways

The way in which educational practitioners utilise the available theories and research in higher education can be paralleled in the progress in the student. When formal education begins it starts with simple repetitive drill type exercises requiring the development of respondent behaviour. Respondent behaviour is defined as a direct response to a stimulus [23]. Many of the computer based instruction packages produced, both in their early development and for this level of education, require this form of overt response from the student. Computer based instructional environments, in their nature require some form of direct interaction or response from the student to be able to function in a satisfactory manner if at all [19].

The two most obvious educational theories that have to date become an integral part of computer aided learning are, reinforcement and the cognitive style/learning strategy coupling. It is important for student development and learning to have some form of reinforcement of what is being learnt. At present, in higher education, the highest governing factor is the final mark or grade in an assessment task. If a question is asked on a topic this in itself gives positive reinforcement to its importance. Also the ‘correct’ answering of questions, along with the ‘wrong’ response, directs students to work that has either been successfully learnt or that which has not [23, 19]. Mark and grades also have a motivational effect on students. Marks alone are not the total motivation for most students. The presenter and actions of a tutor can have profound effects upon student performance, especially if they are encouraged to take greater control of their own education. Tutors can also more easily provide the implicit knowledge students require to truly master each topic [24].

The emergence of the importance of cognitive style has shown itself both in the way education is conducted and the awareness of students as to how they learn. Yet “cognitive style needs to be distinguished from learning strategy. In the present context, a style is considered to be a fairly fixed characteristic of an individual, while strategies are the ways that may be used to cope with situations and tasks. Strategies may vary from time to time and many will be learned and developed. Styles, by contrast, are static and relatively in-built features of an individual. Cognitive style can be defined as an individual’s characteristic and consistent approach to organising and processing information” [25].

The identification of various cognitive styles for the producer of computer based instructional material can be seen in both the organisation and presentation methods used, often enabling the students to select amongst various representations dynamically, and so maximising their learning effectiveness. This has also caused a greater separation between material and presentation, increasing the degree of flexibility for producers (and tutors) to update both independently. This becomes increasingly important, especially in vocational training, as the half-life of course content decreases [24,5]. (The half-life of material is defined as the time during which at least half of the content taught continues to be relevant and useful).

Increasing the flexibility of material personalisation and interaction, increases the student's ability to deal with the cognitive load required to both learn and integrate new information. Material with a higher intrinsic difficulty level should be studied in smaller amounts, thus reducing the extraneous additional cognitive load the organisational design places upon the student. This lighter overall cognitive load enhances the process of moving knowledge from the relatively small capacity of short term memory to the larger long term memory [26]. The environment in which the acquired knowledge is to be used also adds to the cognitive loading of students. In vocational training especially this environment, along with the associated facts, and increasing the participatory nature of learning has proven to yield greater success in the transfer of knowledge from the learning to work environments [27].

Vocational training has led the way in the adoption of competence as a measure of learning in terms of the actual assessment methodologies applied. This has led to the development of standardised meanings, dependent upon task and context, for competence and its acceptable levels. It has also caused a challenging of the historically held norms of academic assessment. The assessment and the teaching methods to be used are increasingly being derived from answering the questions; 'What are we trying to achieve?', and 'What is it for?' [5]. The attitudes of students also play their part in the preparation for assessment and the 'ease of use' of the teaching medium. For example it is not appropriate to test a computer based topic with pen and paper. It is often not appropriate to assess competence levels without the 'tools' with which the student is required to conversant, for example the operation of a lathe. It is equally important, and especially if the assessment is to be computerised, that students become familiar with the 'tools' during the learning phase and that the training integrates the use of the 'tools', the computer, into the total programme [28]. If the usage of the associated technology is not taught within the learning phase then it can come between the student and the material itself reducing the probability of learning taking place.

Even though educators have strong motives to use assessment to test students competence, students can defeat this through their desire or need to pass rather than learn. T H Huxley's acid criticism of the examination system as it was in England a century ago as reported in [3], gives an everlasting reminder of the purposes of and attitudes towards assessment. Examinees, he said, work to pass, not to know, and nature takes her revenge: they do pass and they don't know.

With the increasing need for flexibility in education, the student or learner is required to take a more active part in the process. The more students participate in their

learning and the greater control over the learning process they have, increased interaction within the educational programme is required. The more a student is involved in the process of education and especially the assessment of what has been learnt, increases the probability that material will be actually understood to the agreed level of achievement [29]

2.4 Structure of programmes, potential uses.

“Many researchers have noted that an essential prerequisite for efficient computer assisted instruction (CAI) is a system’s ability to match individual learning needs to appropriate learning activities” [30]. A computer based educational system needs to integrate existing knowledge of the domain concerned with a model of what each student does and does not know, and the learning styles and preferences of the student to enable the best selection of teaching strategy. Selection of a strategy is required to be both accurate and dynamic allowing sufficient flexibility to accommodate the changing needs of each student. In the past “typically an ITS (intelligent tutoring system) is based on a convergent type of learning: the student is assigned tasks from a small domain, eg adding fractions. This narrow design of task structure gives rise to problems of student motivation” [30]. It has been said that not until students go to “school is knowing uncoupled from doing” and this fact has been blamed for the failure of some students to demonstrate sufficient understanding [31]. Utilising open structured CAI this recouping can be achieved by embedding material/tasks to be learned into explorative environments and games. Many younger students, especially in the primary school age group, learn more by playing games within which their knowledge must be applied, than by just being able to repeat facts or perform tasks on demand. For older students and at tertiary level games are usually replaced by situated learning, where learning takes place in the situation in which the knowledge is to be applied. This can easily be seen in trade training; for example a mechanic learning piston settings, as this task requires both physical and mathematical skills to accomplish properly.

When the desire is to implement a more complete computer mediated educational environment, or ‘virtual classroom’, the interaction of both student to tutor and student to student needs to be considered alongside that of student to material. The use of a virtual classroom enables there to be greater flexibility on time and location of study. Facilities for both asynchronous (student and tutor are not present at the same time) and synchronous (student and tutor are present at the same time) learning and interaction need to be present in such an environment. Extensive use of electronic mail which is primarily text based, facilitates student to tutor and student to student communication and often yields a greater level of interaction than in a face-to-face environment. Conferencing and document sharing facilitates the use of group based projects as it enables several students to simultaneously view and interact with the same material and each other. As with each new and different way of learning, students and the producers of computer based educational environments need to learn how to maximise the benefits and minimise the limitations of all the mediums used for interaction and presentation [32].

Creating new educational environments poses many challenges to the designer. It can be seen from past experience that “most multimedia programs fail because they

merely add video and graphics to page-turning programs. Creating effective multimedia programs means taking seriously the idea of learning by doing. Good educational software is active not passive, and ensures that users are doing, not simply watching” [31]. Utilising multimedia, as apposed to single media, to present material can often add degrees of complexity to the design. Its use will, by necessity of its nature, instigate a rethink of overall programme design. The uses of a computer in the educational process range far beyond the established ability to provide individual, and individualised, instruction. With good design and implementation of appropriate technologies computers (1) present users with tasks that interest them, (2) allow users to be inquisitive and explore, (3) offer users the possibility of recovery from failure without embarrassment, and (4) put learners in control of the task [31].

A well designed computer based educational environment has much to do with educational theory, as the matching and the use of appropriate technologies for both student usage and the type and style of presentation. Information and telecommunication technologies can provide new opportunities, and enrich existing teaching possibilities. Blackboard, overhead projectors, audio and video tapes are some of the physical tools educators have become increasingly accustomed to using. Particularly in a university environment small group discussions, simulations, case studies and individualised criteria have been added as instructional strategies that can be used as tools to enhance teaching and learning. The use of multimedia is capable of producing an environment in which all of the tools mentioned above can be utilised, yet can also facilitate the effective incorporation of various instructional strategies. “Instructional systems design provides a set of activities to guide instructors in developing appropriate teaching methods. This set of activities consists of:

1. Identifying the instructional goal
2. Examining the educational setting
3. Identifying learner characteristics and entry behaviours
4. Specifying objectives for the learner
5. Anticipating a method to assess achievement of objectives
6. Developing an instructional strategy
7. Designing and developing instructional materials
8. Designing and conducting evaluations of the instruction” [7]

This design methodology is applicable to any form of instruction and is especially useful when implementation is to incorporate computer mediated educational aspects.

The applicability of the different technologies and their configuration will depend on the situation of the learner (home based, company based or in a training centre) and on the content of the training (problem solving or interpersonal/communication oriented). ‘HOW’ to learn is defined by ‘WHERE’ the learning is taking place and ‘WHAT’ is to be learned” [1]. To achieve the desired level of flexibility in the design and application of educational systems, all forms of presentation and interaction need to be considered - human, computer, information. “Intelligently designed computer programs can make a significant and positive difference in the way we teach children and train adults. Such technology will engage people’s natural inclination to learn” [31]. The true ‘targeted results’ from such systems is stable student knowledge [30]. This is where the student has the correct knowledge and can demonstrate its correct application.

2.5 Barriers to change

There are many things that make us reluctant to change, including the unwillingness to change at all. The adoption of computer mediated education, and computers in education in general, has and is facing many barriers. The abilities and understanding of educators to use and utilise the technology can relegate a computer to a corner of the room used by enthusiasts only [11]. Added to this is the fact that the knowledge and experience gained by the establishment of educational programmes utilising the technology is not often shared amongst institutions [1].

For those willing to invest in technology for educational purposes the acquisition of the hardware required can be a daunting task, even though the finances are available to do so. Inadequate resources for housing computers, the required technical support, and the lack of appropriate software systems will render any technology effectively useless [11, 1]. Many educators are unwilling to use the software applications that are available because they perceive

1. no direct applicability to their students
2. a lack of student monitoring, of progress and understanding
3. an inability of systems to actually help in achieving desired educational goal
4. difficulties concerning the validity of learning methods used and their effectiveness
5. its effects, or ineffectiveness on results in standardised testing [1, 22, 33].

Another problem are the limitations of hardware, software and communications facilities to achieve the desired interaction, or even what the advertising has promised, can reduce the use of computers to trivial tasks [1, 33].

For the student, the lack of physical access to available computers, coupled with a lack of prior experience can be a serious demotivating factor [11, 1, 34]. Once actually using computer based educational systems a student can get lost in a sea of material with no real direction to achieve the desired learning goals, and/or lose sight of the overall picture. Poor structure, the lack of clear paths through the material, inaccessibility of some information and the lack of an overview can reduce good programmes to being merely information provision systems rather than achieving any real level of student understanding of the material. This is especially so for hypertext and internet based programmes [46].

The level of prior knowledge and experience of computers and other related technology can seriously effect a student's ability to fully utilise computer based learning environments. They require a certain level of support to enhance their usage and acceptance of computer based learning as a valid and effective method. Increased understanding and motivation coupled with good design enables students to interact and support themselves, both in terms of understanding and the use of the system, often increasing the responsiveness of technology shy students to participate fully. This is especially so when design overuse of one medium, particularly multimedia, can confuse and frighten off some students [1, 22, 34].

Any educational programme can be forced into abandonment due to the environment in which it must operate. This environment is the responsibility of the educational providers, to ensure that it fosters, not harms, educational development and student learning. Institutions do not exist in isolation and must cope with the pressures placed upon them by external sources. "There are three main environmental forces acting on systems of higher education. One is customer demands: the pressure of students to

get into colleges and universities and to pursue the curricula they want when they get in. A second force is manpower needs: the 'suction' drawing graduates into employment, and therefore influencing criteria and certification. The third force is patron's influence: higher education systems are not (they never have been) supported by customers or employers; they are nowadays under the patronage, that is, the ultimate financial control, of the state" [15]. In New Zealand the government is still the major, but by no means the only, single financial contributor to higher education. New Zealand universities are also governed in their operation by law, as they exist by act of Parliament, usually one for each institution.

2.6 Evaluating the teaching and the learning

For education to achieve its aim of teaching and the students of learning, both the teacher and learner must be aware of what the learner already knows and what is still to be understood. This is the role of assessment. Assessment refers to the judging the amount of learning that has taken place for each student, where evaluation is the determination of the effectiveness of the teaching programme. Evaluation and assessment can be conducted formally, informally and at any time throughout the course. Some of the more effective assessment in determining true understanding is that done when the student is unaware that it is taking place, or has been integrated into the course in such a way as to minimise the 'test stress' that can mask true understanding. "The assessment becomes very much an integral part of the learning. There are a set of key questions that need asking early in the process:

- What are the purposes of the learning experience?
- What does the learner think are the purposes of particular work that has been agreed or allocated?
- What are the purposes of the assessment procedure that has been agreed or negotiated?
- What is the relationship between the learning experience and the assessment. (Has the assessment become the whole organisation of the experience?) [29]

There are times at which the student is not a party to the setting of the assessment. This is most often seen when it results in a certification of performance to a set standard level, as in national examinations or trade certification. Independent of the type of assessment that is to take place it always should be matched to the type of studies undertaken, style appropriate to the task, level of understanding set at the beginning of the teaching phase, and what the examiner wishes to know about the level of learning of the students. "There is a vast range of types of assessing available. From informal and casual observations, through teacher organised and marked questions, standardised tests, to assessment criteria devised and used by the learners themselves. All have a place in facilitating learning, but not all may be realistic or desirable in any particular situation. Like all learning/teaching activities, assessing needs to be carefully chosen to suit the intended purposes, expected outcomes and constraints" [29].

Evaluation of the teaching itself should also be undertaken throughout the entire process in an effort to determine the effectiveness of each component part and where

improvements can be made. "Evaluating during the learning process (often termed formative evaluation) is important for learners and teachers alike, aiding the teacher in fine-tuning (or hole plugging and patching) and the learner in becoming more involved in, and aware of, the learning process. In addition, the communication which should arise naturally from such evaluation is a vital aid to the learners and teachers becoming more aware both of themselves and each other's views, problems and expectations. It should be a sharing process. Evaluation subsequent to learning (often termed summative evaluation) is necessary for providing information to both learners and teachers about learning which has occurred, perhaps comparing it with that which was intended. Factors enabling and constraining the learning process may also be identified. Such evaluation can provide the learner and teacher with information about what learning is still necessary to achieve particular goals, and the teacher about development of future courses" [29].

It is important to evaluate all parts of a course not just the newer aspects. This allows for the continual identification of areas for improvement. It can be especially so with types of delivery with which the presenter is very familiar, thus reducing the tendency to present the 'same old material in the same old way'. Newer aspects of the teaching programme are easier to identify, although not always to evaluate. This can be particularly so when introducing the use of multimedia into the learning environment. "It is important to evaluate multimedia instruction at least partly because the technique is so new that most instructors lack experience in what is or is not effective. Formative evaluation assists faculty members in providing the best possible multimedia instruction that is effective in terms of achieving the identified educational goals. By focussing on what type of multimedia instruction most closely achieves specific kinds of educational goals for students with particular characteristics, its appropriate role can be more clearly identified" [7].

2.7 The education of technologists and engineers

In the fields of engineering and technology there is a prerequisite requirement to use, teach and utilise new technology. This is both in terms of what is being taught and how it is taught. In the past educational institutions have kept up at a reasonable level with developments in what is being studied while not neglecting the 'basics' of the physical laws on which they are built. Unfortunately the developments in educational theories as they relate to teaching in any field, have not been adopted by engineering schools in the same way [35].

The education of technologists (engineers and technologists can be taken as interchangeable terms throughout the text) involves many complex domains and domain interactions. It is no longer possible to teach each aspect in isolation, but the interrelations between fields of study must be recognised and included within the programme of study as a whole. This includes the incorporation of the prior knowledge, or lack of it, that each student brings with them, and the allowance of adequate opportunities for the restructuring of this knowledge in the light of new information. "People restructure their knowledge in the course of learning in a way that is analogous to conceptual changes in the history of science. Students bring conceptual systems to learning, and instruction fosters an evolution from untutored theories held by beginners to more sophisticated theories and conceptualisations. The

more powerful theories, by which competent students organise their knowledge, enables them to interpret problems in ways that beginners do not. Recent research contrasting novices' and experts' performances had shown that conceptual shifts from surface-level problem representations to deep-structure representations occur with the emergence of expertise in a field" [36]. The transfer from surface to deep learning, from the knowing of facts to the understanding of principles, is not only necessary in the field of engineering but is mirrored in aspects of many learning taxonomies that have been developed by educational researchers, for example Bloom's taxonomy [47]. From the students' perspective the shift is from knowing as much as is necessary to pass the exam and no more (surface learning), to knowing not just the 'what' but also the 'why' and 'how' (deep learning) [37].

The development, and continual adjustment, of the total engineering syllabi to ensure the 'production' of qualified graduates and accommodate the challenges being faced in today's educational and industrial environment is no small task. One suggestion is to modularise the course so that each section can be presented in the best possible manner. When courses are modularised, especially if given at separate institutions as is the proposal for a European degree system [16], is that it is very easy to lose focus on how each section contributes to the whole, ie the big picture. Modularisation is particularly attractive when dealing with large numbers of students. However according to one engineer "consequence of modularisation is the increasing tendency to place responsibility for student performance upon the educator. This inevitably leads to an overemphasis on examination success as opposed to an understanding of the subject and may also result in student apathy. The responsibility for performance should remain with the student who, with guidance, should develop other responsibilities associated with a professional career. Lecturing should not primarily consist of training to pass examinations, which some see as the escape from the limitations placed upon us. Careful design of engineering syllabi will free the student from the restrictions of the lecture theatre and guide him or her towards the exploration of an exciting, ever-changing subject. A major implication is the necessity to provide well designed laboratories and high-quality extracurricular resources such as library books, videos, and computers" [38]. The modularisation of the engineering and technology courses is one implementation of the desire increase in flexibility within the degree. Specific modules can be taught by experts in the field, thus drawing on each institutions particular research areas and upon the specialities of others. The modularisation of a course places greater emphasis on good structure and introductory sections. When a course is broken up into many units there is a greater risk of students not being able to form the inter-relationships between the units, being overwhelmed by the depth to which each topic must be understood. The breadth and depth issues in course design are universal and not limited to those which have been modularised.

As mentioned earlier the first year of any course of study is often the most difficult for students as they seek to adapt the knowledge they already have to their studies and the acquisition of other skills. A lack of the 'basic skills' needed as a professional engineer that are not strictly related to any one discipline, but applicable to all, is being addressed by the University of Edinburgh by incorporating them into its first year courses. "There is an acknowledged need for engineering students to acquire basic transferable skills to be effective as practicing engineers in industry. Often it is the abilities to communicate with other people and to organise time and work which

mark out the graduate who is preferred for employment. Many engineering departments have modified their courses to enhance these skills so as to produce graduates who are able to step directly into positions of responsibility. The greater need, however, is when students arrive at university. At this stage, the majority are coming with little or no experience of the freedom which university life affords them. At the same time, they are suddenly faced with a vastly different educational environment in terms of the way in which knowledge is delivered and the amount of independent learning which is required" [39]. By making these aspects of the course compulsory and taken within the first year of study, the university is seeking to both bridge the gap between the systems of education students are familiar with and university study, and to prepare the students for their future as professional engineers.

Another issue evident in an engineering education is the acquisition and proficiency in practical tasks, for example the operation of specialised machinery. One member of staff at the University of Edinburgh has found that it is possible to combine the best of both worlds in terms of learning in an apprenticeship or university environment, by careful construction of the course. "During the construction of several labour-intensive high-technology projects my research group recruited a number of school leavers and vacation students. We found that people with a high motivation working with more experienced engineers on what they regarded as an important project could acquire and retain a wide range of theoretical and practical skills in an astonishingly short time. It seems to be important that people should see the entire sequences from problem identification, to analysis, solution synthesis, detail design, manufacture, test, modification and redesign until satisfactory performance is achieved. The project assignments should be just, but only just, within the ability of the student. Impediments and restrictions of access to materials, components and technical information should be minimised. Help from seniors should be easily available but it is also desirable that students should be giving help to and supervising the activities of juniors as soon as possible. ... The goal should be the maximum performance of the product rather than acceptable examination marks" [40].

An attempt has been made to map the components of an engineers education. This is shown in Figure 2.1 where "the oval enclosures contain aspects of an education that are special to engineering, the rectangular boxes are intellectual activities that have a bearing on engineering and the round-cornered boxes alludes to personal attributes" [41].

2.8 The impact of technology on what is taught

Continuous improvements in technology have caused, in turn, continuous changes to be made in terms of what a technology or engineering degree contains. If new material is to be introduced, is the course to be made longer, or is other material to be removed from the course, or is there to be a reduction in the depth to which topics are taught? The inclusion of the computer as a tool for managing information and the execution of those tasks that are too complex to be completed 'by hand', for example numerical calculations and the analysis of large amounts of data, has resulted in both the ability to implement areas of theoretical knowledge that were not previously possible and the introduction of the study of the computer itself. When the study is not limited to the computer alone but includes other related issues it is referred to by the name Information Technology (IT).

"Engineering practice has changed radically over the last decade as a result of advances in computer-based tools for design, analysis and simulation in the one hand, and for management, office technology and production control on the other. Yet many (if not most) European engineering curricula remain firmly wedded to a traditional, highly mathematical view of their subject rooted in applied science. Moreover, students, graduates and industrialists continue to criticise the relative weightings given in such curricula to foundation learning, specialist subject knowledge, and generic transferable skills. Applications of IT in engineering education tend to fall into three categories: purpose-built CAL simulations and exercises; generic software of various types; and more specialist commercial engineering packages. To these should now be added the use of computer-mediated communication (CMC) as part of the general academic and educational environment. Unfortunately, even when such IT (information technology) tools are adopted, it is all too often simply as a new way of carrying out old exercises" [42]. Information technology like all other technologies can be both studied in its own right and used as a medium for study in other areas. It should be used to supplement existing course material, used to extend the learning opportunities for students and staff alike, and develop new creative ways of addressing problems and issues. Information technology, as stated above, is often used as an automated way to perform exercises previously done other ways, usually by hand. The development of more 'user friendly' interfaces, computer conferencing, and the internet to give just two examples, have become much more accessible and easier to use. This in turn has a circular effect, increasing the amount and level of material available.

The use of the internet and conferencing, both video and computer conferencing, has enabled more people to access more information and to interact with others not in the same physical location than ever before. The effects of this on education can be overwhelming. Institutions can offer courses 'globally' and also are able to educate students 'globally'. As a student you are able to 'pick and mix' your courses, to pick the course which best meets your needs wherever and whenever it is offered. This in turn increases the competition between educational institutes, forcing them to specialise more making a niche market for themselves which then offers more choice to perspective students. One specific area that has been eagerly utilising these developments is the continuous professional development of engineers. With courses being made available via the internet those in the workforce may access material and courses without interrupting their work in the same way a residential programme

does. A course has been developed at Dublin University to offer graduate studies in machine vision. The online material is also supported by a dedicated electronic discussion group, interacting via EMAIL [43].

The study of engineering is often perceived as an investigation into how things work, ie the nuts and bolts, and the development of new things, more nuts and bolts. This view ignores the people element in all fields of engineering. "The human element in technology is no frill. As machines become more sophisticated the human problems of technology become more challenging" [3]. The incorporation of the human aspects of technology into the programmes of study is what will carry engineering into the future. This is reflected not only in the understanding of the 'people factor' in design but also in terms of how engineering based subjects are taught, fostering interaction between students and the student and tutor. The study of the human aspects of technology and its use, should not be 'tacked on' to a course but be integrated into each subject. This encourages the student not to treat the human element as a separate issue. Staff and students of engineering alike often find difficulty in the teaching of, developing competencies in, and the assessment of understanding of these so called 'soft' issues. They cannot be measured with a ruler, there is not one 'correct' answer and require individual interpretation. The lack of a single concrete answer contributes to the difficulty that most engineering students find with this type of subject and is more often reflected in its lack of popularity among the students in comparison with their other subjects.

2.9 Concluding comments

In education as in life, things are constantly on the move. Nothing is constant but change itself. This is reflected in the speed at which technological developments are highlighting the shifts in educational theory and practice. The true core principles of education remain constant, that is the development of knowledge and understanding to the benefit of society at large. The physical locations of tutor and student which were once vitally important, are becoming increasingly transparent. Technologies are available to link people from disperse locations and enable the equivalent interaction to take place. This has resulted in a rethink of many theories of the place of interaction within education.

As society and its needs change, so must our educational structures, processes and institutions. It is possible to retain aspects of existing approaches while not ignoring new developments. The challenge facing educators today is to become multi-skilled and to adopt a team approach to learning. Tutors require skills and expertise in their specific disciplines, knowledge and understanding of the appropriate theories of learning, and to be adept at handling any associated technologies. The team approach involves the tutors, educational experts, and technology specialists as would be expected. It also includes the students giving a complete perspective on the learning experience. As a by-product of this approach the student is encouraged to become and remain an active participant in the learning process.

2.10 References

1. Lieve Van den Brande
Flexible and Distance Learning
John Wiley & Sons Ltd, 1993
2. G W Skates
Partnerships between industry and higher education
IEE Engineering Science and Engineering Journal, Vol 4 No 3, June 1995
3. Eric Ashby
Adapting universities to a technological society
Jassey-Bass Publishers, London, 1974
Chapter 3, Technology in Education
4. Lynne Reder, Robert L Klatzky
The Effect of Context on Training: Is Learning Situated?
Carnegie Mellon University, Human-Computer Interaction Institute, Technical Report CMU-HCII-94-108
5. Andrew Gonczi (Editor)
Developing a competent workforce
National Center for Vocational Education Research, 1992
6. Danny Saunders and Phil Race (Editors)
Aspects of Education and Training Technology XXV
Developing and Measuring Competence
Krogan Page, London, 1992
7. Dennis R Falk and Helen L Carlson
Multimedia in Higher Education
A Practical Guide to New Tools for Interactive Teaching and Learning
Learned Information Inc., Medford, New Jersey, 1995
8. Urban Dahllof, John Harris, Michael Shattock, Andre Staropoli, Roeland in't Veld
Dimensions of Evaluation in Higher Education
Jessica Kingsley Publishers, London, 1991
9. Michail Russel, Walt Haney
Testing Writing on Computers: An Experiment Comparing Student Performance on Tests Conducted via Computer and via Paper-and—Pencil
Educational Policy Analysis Archives, Volume 5 Number 3, 15 January 1997,
<http://olam.ed.asu.edu/epua/v5n3.html>
10. Urban Dahoolf, John Harris, Micheal Shattlock, Andre Staropoli, Roeland in't Veld
Dimensions of evaluation in higher education
Jessica Kingsley Publishers, London, 1991
Chapter 6, Towards a New Model for the Evaluation of Teaching: An Interactive

Process-Centred Approach

11. Jonathan Pemberton, Joanne Smailes
Educating the Masses - Is IT a Tricky Business?
The Computer Journal, Vol 36, no 4, 1993
12. Don Menn
Educational Multimedia, Holy-er than thou?
PC World, October 1993
13. Don Menn
Multimedia in Education, Arming our kids for the future
PC World, October 1993
14. Peter W G Write
Industry and Higher Education
Collaboration to improve students' learning and training
Open University Press, 1991
Part 3, Graduates and the Needs of Employment
15. Eric Ashby
Adapting universities to a technological society
Jassey-Bass Publishers, London, 1974
Chapter X, Mass Higher Education
16. M Woolf
International education: pitfalls and possibilities
IEE Engineering Science and Education Journal, Vol 4 No 1, Feb 1995
17. *Teacher in a box, the debate over multimedia learning*
PC World, October 1993
18. Reda Abouserie, Dennis Moss, Stephen Barasi
Cognitive Style, Gender, Attitude toward Computer-assisted Learning and Academic Achievement
Educational Studies, vol 18, no 2, 1992
19. Kenneth Silverman, Ogden R Lindsley, Kathy L Porter
Overt Responding in Computer-Bases Training
Current Psychology: Research & Reviews, Winter 1990-91, vol 9, no 4
20. Ann Hill Duin
Computer Documentation - Centering on the learner
Journal of Computer Based Instruction, Spring 1990, vol 17, no 2
21. Richard Riding and Graeme Duoglas
The effect of cognitive style and mode of presentation in learning performance
British Journal of Educational Psychology, vol 63, 1993

22. C H Morgan, I D Lilley, N C Boreham
Learning from Lectures: the Effect on varying detail in lecture handouts on note-taking and recall
Applied Psychology, vol 2, 1988
23. Fred S Keller
Learning: reinforcement theory
Random House, 2nd Ed 1969
24. Michael L Burger, John F DeSoi
The cognitive apprenticeship analogue: a strategy for using ITS technology for the delivery of instruction and as a research tool for the study of teaching and learning
International Journal of Man-Machine Studies, June 1992, vol 35, no 6
25. Richard Riding, Wugene Sadler-Smith
Type of Instructional Material, Cognitive Style and Learning Performance
Educational Studies, vol 18, no 3, 1992
26. John Sweller
Some Cognitive Processes and their Consequences for the Organisation and Presentation of Information
Australian Journal of Psychology, vol 45, no 1, April 1993
27. Farhad Analoui
Training and Transfer of Learning
Aveburg, Ashgate Publishing Ltd, 1993
28. C Frasson, M Kaltenbach, J Gecsei, J-Y Djamien
An Iconic Intention-Driven ITS Environment
Intelligent Tutoring Systems, Second International Conference, ITS '93, Montreal, Canada, June 1992 - Proceedings
29. Duncan Harris and Chris Bell
Evaluating and Assessing for Learning
Kogan Page, London, 1994
30. Yoneo Yano, Akihiro Kashihara, William McMichael
Stabilising student knowledge in open structured CAI
International Journal of Man-Machine Studies, Nov 1992, vol 37, no 5
31. Roger C Schank
Active Learning through Multimedia
IEEE Multimedia, Spring 1994
32. Starr Roxanne Hiltz
Correlates of learning in a virtual classroom
International Journal of Man-Machine Studies, vol 39, 1993

33. Robert Bruce
Creativity and Instructional Technology: Great Potential, Imperfectly Studied
Contemporary Educational Psychology, vol 14, 1989
34. Michael Cronin, Karen Cronin
A critical analysis of the theoretic foundations of interactive video instruction
Journal of Computer Based Instruction, Spring 1992, vol 19, no 2
35. V John
A future path for engineering education: Educating engineers for Europe
IEE Engineering Science and Education Journal, Vol 4 No 3, June 1995
36. Robert Glaser
Learning theory and theories of knowledge
Learning and Instruction, European Research in an International Context, Vol 1
Erik De Corte, Hans Lodewijks, Roger Parmentier, Reiter Span (Editors)
Pergamon Press, Oxford, England, 1987
37. *The Virtual Seminar Series*
Monash University, Churchill, Australia, Oct 1994 - Jan 1995
38. C J Moore and K M Holford
University degree courses: producing the kind of engineer that industry needs
IEE Engineering Science and Education Journal, Vol 4 No 1, Feb 1995
39. G N Blair and C M Robinson
Professional skills for first-year engineering students
IEE Engineering Science and Education Journal, Vol 4 No 1, Feb 1995
40. S H Salter
Some new heresies on the training of engineers
IEE Engineering Science and Education Journal, Vol 4 No2, April 1995
41. W T Norris
A map of an engineer's education
IEE Engineering Science and Education Journal, Vol 4 No2, April 1995
42. Christopher Bissell
Revitalising the engineering curriculum: the role of information technology
IEE Engineering Science and Education Journal, Vol 5 No 3, June 1996
43. Paul F Whelan
Remote access to continuing engineering education (RACeE)
IEE Engineering Science and Education Journal, Vol 6 No 5, October 1997
44. Gordon Dryden & Dr. Jeannette Vos
The Learning Revolution
Profile Publishing Ltd., Auckland, New Zealand, 1993

45. P A Witting
Trends in engineering education
IEE Engineering Science and Education Journal, Vol 5 No 2, April 1996
46. Nick Hammond, Lesley Allinson
Extending Hypertext for Learning: An Investigation of Access and Guidance Tools
Department of Psychology, University of York, England
47. David R Krathwohl, Benjamin S Bloom, Bertram B Masia
Taxonomy of Educational Objectives
The Classification of Educational Goals
David McKay Company Inc., New York, 1956

3 Review of Existing Technological Systems and System Components for Flexible Education

There is a varied array of technologies available to the educator. Some are more specific to implementation settings than others, yet used appropriately all can assist in the educational process. This chapter outlines some of the types of existing systems and their related components. Where available an example of the implementation of each system is given along with an evaluation of its effectiveness. Varying levels of detail are presented which reflect some of the complexity and number of considerations that need to be made when such systems are implemented.

3.1 Interaction and interfaces

The interface to any system, which the user sees and interacts with, will often dictate its usefulness and useability. The components of a system with which the user interacts comprises both the software and hardware. Is the screen divided up into sections or are multiple screens used. Is interaction via touch screen, voice commands, mouse, keyboard, or a combination of these. What physical size benefits/limitations are present.

Each educational programme or environment, has a specific target in terms of audience and purpose. This often dictates the type and amount of interaction as well as the style of material presentation to be used. The material itself is another key to its presentation. The overall goal of any system is to achieve a predefined set of learning objectives. The technology must be designed and implemented such that it assists the achievement of these learning goals. As stated in Chapter 1 this means that the objectives of the technological goals must be subsumed in the learning objectives.

One of the key problems of the application of technology in educational programmes is that learning the technology as well as learning the subject can cause cognitive overload for the student. Cognitive overload can be avoided, or at least reduced by the breaking up of the material into small sections. Sweller [1] addresses this in the following way. "The difficulty of some tasks can be increased by the way they are presented. Material that is less intrinsically difficult may be presented in larger segments, but material that is intrinsically difficult by necessity must be presented in a simpler manner. Material should be structured and presented in such a way as to enhance the way the subject is learned thus reducing the extraneous cognitive load". The use of diagrams, simulation, animation and sound also reduces the overall cognitive load by stimulating multiple senses giving the student greater exposure to the material without corresponding increases in loading

An example of a well constructed system is the trouble shooting simulation for diagnosing and repairing faults in electronic devices created at Université do Montréal, Canada [2]. Figure 3.1 [2] shows the screen layout the student is presented with. The designers of this system were seeking to represent the reality of the working situation while restricting the student to a logical order of activities. "The student is required to plan each action, stating reasons and objectives, before it can

be carried out. This simulated environment consists of three windows, one showing the device itself and can be interacted with as if it were real; one showing a diagram or schematic of the device at the logic level which is linked to the physical representation so that interaction with one is reflected in the other, and a window where the student gives reasons for their actions. An initial fault is presented in symptom form and the student must state an original hypothesis of cause, decide what test and measurements to carry out, revise the hypothesis using the results until the fault is identified and the correct solution stated” [2].

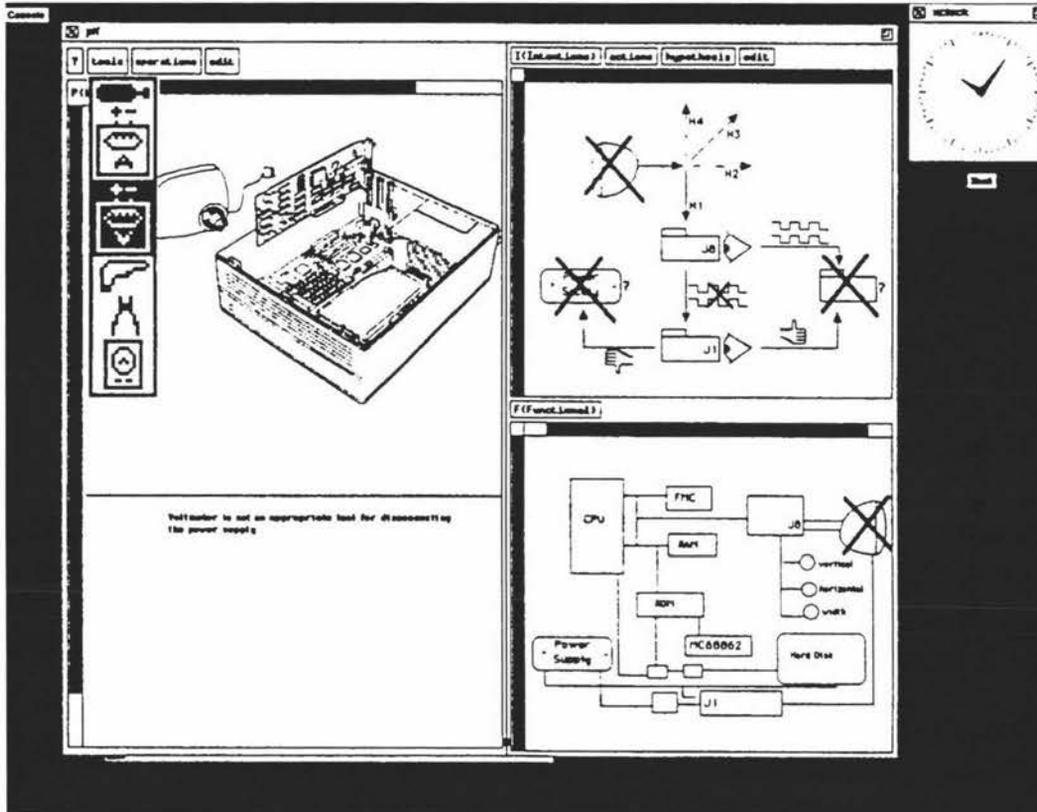


Figure 3.1 Screen layout depicting areas of interaction and information presentation. The tool selection menu is currently active.

This system is graphically based being driven by the manipulation of icons, eg the representation of tools by their pictures. Here the process of learning, (to diagnose faults, propose hypothesis and plans of action) is partially evident to the student. Especially with the use of computers the learning process can be made visible or invisible to the learner. Systems with visual learning components make the learning process more visible and therefore, can achieve a greater student understanding of the learning process itself not just the material [3].

Active learning is promoted within the simulation by requiring the student to make direct responses to the system. It can also be seen that feedback is immediate and includes the information that would be present in the real world environment [4, 5].

The use of more graphical interfaces rather than text only has lead not only to greater possibilities and ease of handling by the user, but has also raised interesting issues for the developer. Handling both the action or event driven interactions (eg the

pushing of a button), and the continuous actions (eg mouse movement, the playing of sounds), require a greater degree of synchronisation. The underlying computer system's ability to handle this synchronisation will affect the fluidity of what the user sees, and the response time to any actions [6]. The level of synchronisation becomes increasingly important when use is made of multiple presentation formats, to stimulate multiple senses of the user, especially where these are simultaneous. This is often done for 'realism' and to make the student feel a part of the environment [5].

The use of object-oriented authoring environments and languages is one solution to the issues of synchronisation. Many of the structures required are inbuilt into the programming metaphor itself, and so relieving the author of the concern for such issues at greater depth. There are also a number of systems available to assist the author in the process of creating and managing graphical user interfaces. Irrespective of the tools selected there are a number of issues which are "fundamental to the successful implementation of any multimedia instruction scheme, (i) an appreciation for the central integrating and coordinating role of the computer, (ii) the realisations and acceptance of the fact that many kinds of instructional resource need to be used, (iii) the optimal integrated deployment of the resources that are available for use, and (iv) the production of multimedia resource pack" [7].

With an increased use of graphics for material presentation and student interaction with an environment, associated help facilities becomes increasingly important. This is especially so in the case of context sensitive help facilities. Such systems should provide assistance both in the understanding of the material and the operations and interactions with the environment [8]. Guides to the operation of the environment often run independently of, yet joined with, the main instructional parts of the environment. This flexibility allows the student to refer to operational instructions while engaged in viewing the material without having to stop one activity to engage in the other.

3.2 Authoring environments and tools

No matter what type of display and transmission mechanism is used, the material needs to be created first. In relation to multimedia this process is called authoring. There needs to be a degree of matching between the authoring and delivery environments to ensure that what is created can be delivered. There is a large number of possible authoring environments and methods from which to choose. "Often, even though an objective approach may be used, final system selection is likely to be by both personal likes and dislikes, and by preferences to a particular manufacturer's equipment. Political and financial factors are also likely to influence system selection" [7]. A check list of possible questions and considerations when choosing an authoring environment can be found in Appendix A.

In general terms an authoring environment should [5, 9, 10, 11]

- reflect the requirements of the final system, eg can the handle presentation of material and interactions required,
- be able to utilise the functionality of the implementation environment,
- be adaptable to changing needs,
- be able to sequence and construct presentations, dynamically adjusting the levels of complexity required,

- be simple to learn and use,
- have good and consistent navigation, and
- follow international standards for communication and interaction.

There are many authoring environments, languages and construction tools. Some programming languages have been specifically designed for authoring computer aided learning (CAL) material, others have been used for this purpose, ie more generic languages. Non CAL specific languages include APL, Basic, FORTRAN, C, LISP, Pascal, Prolog and Smalltalk. Tools have been created for dealing with specific media, for example AVCAT, CAVIS, CYCLOPS, IVG, PHILVAS, TICCIT, AUDIOTEXT. General development and organisational environments have included ACL, ASSET, CATS, COURSEWRITER, COMBAT, SOMINO, EASYWRITER, FOIL, HAL, IPS, LYRIC, MENTOR, MUMEDALA, PASS, PHOENIX, PLANT, SAL, SMART, TenCORE, Top Class, TUTOR, WISE, XCAL. All of these environments and tools were in use by the mid 1980's [7], and a brief description of each can be found in Appendix B. Since this time authoring environments, for example Authorware^{®1} by Macromedia[®], have been produced that will run on a variety of hardware and encompass the ability to handle multiple material mediums, eg video, slides, interaction, and audio. We need authoring tools due to the complexity inherent in working with multimedia. "If an author has to deal with all the details of multimedia, as is required when using a programming language, it becomes a formidable task" [12].

"The authoring of multimedia information systems consists of the creation of application-dependent multimedia information units, the integration of these units within the application, the construction of information methods whereby users can access this information" [11]. One such system with a metaphor-based development environment is ShareMe - a Shared Multimedia Environment. ShareMe is both a tool for authors who wish to build multimedia information systems (environments), and a run-time systems for these environments. The basic concepts of ShareMe are:

- direct creation of multimedia environments within programming structures
- retrieval of information from multimedia structure spaces consisting of units of text, graphics, video, and sound information
- use of metaphors on the user interface
- offering of a set of navigation methods that support intuitive interaction in the multimedia environment.

Each metaphor used has the same visual and behavioural aspects of its 'real' counterpart. "The benefits of a metaphor-based authoring tool like ShareMe lie in

- easy construction of multimedia environments; the design and implementation of user interfaces are integrated in the tool itself
- the presented user interface metaphors and interaction techniques that guarantee resulting information systems that are easy to learn and easy to use.

It is recognised that metaphors do restrict the types of application that can be constructed using a metaphor-based authoring tool. However, the effect of this disadvantage can be minimised by offering enough different metaphors" [11].

¹ Authorware and Macromedia are registered trademarks of Macromedia , Inc

An interesting metaphor change is evident when moving from a linear text to a hypertext format. Hypertext is the format used for 'internet pages' accessed over the world wide web. An environment was created to be used by students when creating hypertext documents that had restricted capabilities. The students were required to produce a journal paper in both printed text (linear) and electronic (hypertext) formats. The exercise was run several times with refinements to the environments being made each time, arising from the assessment of the documents produced. Additions included increased flexibility within the authoring environment and the incorporation of sound. Problems that were observed in the student hypertext documents included; information in isolation (ie no links to it), too much information on one page and inconsistent navigation [13]. This highlights some of the issues involved in the re-education of the author in relation to the material, and construction and delivery mechanisms.

Other systems have also been developed utilising OLE (Object Link Embedding) and DDE (Dynamic Data Exchange) technologies within a hypertext editor. This particular system has been designed with distance education applications in mind, to increase the flexibility in the development and maintenance of material. It utilises OLE protocols which are standard to all applications to launch and use appropriate applications when desired [14].

Most authoring environments available fall into two camps. One created by an organisation for specific, tailored purposes. The other is the general multipurpose, multiplatform environment. A generalised environment, eg Authorware by Macromedia, will require more learning time on the behalf of the author, but typically have a greater capacity of operations and interactions than their customised counterparts. Most importantly, choose an authoring environment that fits the type of application and the destination system capabilities/limitations, and does what you set out to do. The actual process of authoring material is discussed fully later in chapter 3. Listed below are the major divisions of the authoring process. Different tools will be required during each stage relevant to the formats and requirements of the material.

The process briefly consists of [12]

- concept
- design
- collecting content material
- assembly
- testing
- distribution

3.3 Multimedia materials

Multimedia is an area which is currently receiving a great deal of interest, from educators and consumers alike. When linking together at all of the various products and environments that make up multimedia, it is not easy to get an overall picture of what multimedia actually is. "It is difficult to pin down the essence of multimedia since the term appears in numerous contexts, each with its own nuances. Computer users have multimedia presentations, multimedia workstations, and multimedia databases, but the people involved - the presentation authors, workstation designers, and database researchers - view multimedia from different perspectives. One point

where they can agree, however, is on the essential role played by multimedia data. For instance, multimedia presentations, as their name suggests, involve the presentation of multimedia data, multimedia workstations deal with the processing of multimedia data, and multimedia databases with its storage and retrieval. Perhaps, then, the notion of multimedia data is a unifying thread to digital multimedia [15]. Defining multimedia can be something of a circular argument, but if taking the term itself apart we find *multi* meaning many plus *media* meaning the methods by which information is published. So together they form the way in which information in many forms is conveyed to an audience.

The most common forms in which information is presented are text, graphics or pictures, audio, video or sequences of graphical representations. How these are combined and used is highly dependent upon both the purpose and author of the presentation. Environments have been created that seek to make the author's task easier to manage along with removing some of the implementation details from the direct consideration of the author. These have been described in section 2.2.

When considering multiple presentation forms, the issue of control becomes more critical than for a single media. Controlling the various media is simplified somewhat when each follows standards for storage and form. Standards exist for such things as video and audio compression enabling sequences to be stored more efficiently and retrieved by a number of systems [15]. In conjunction with storage comes issues of access and speed of retrieval. For example, 'When does the storage mechanism need to be accessed so that the material arrives at the appointed time?' [16] This is especially so when two or more forms are to be used simultaneously. An example of controlling various media is in the use of object-oriented dialogue control [6]. In this implementation each type of media are controlled by separate objects to cope with multithreaded events. In this way many actions can be completed and controlled simultaneously to present multiple media material, for example sound and picture at the same time.

A further consideration when using multimedia in an educational setting is the location of the presentation and audience. Is it to be used in a lecture environment, at individual workstations in a laboratory or at home by the students? The tasks of bringing together all the technology, control, and presentation content often fall upon the educator who is often not equipped to fulfil all of these functions. An educator that has found themselves in this position states that "bringing together the components of your MPS (multimedia presentation system) can be a nightmare. Given the challenge of integrating the various parts, you'll want to work closely with representatives from the manufacturers, or the retailer you're buying from. Find out which products do and don't work well together before you buy" [17]. This is sound advice.

For multimedia to be effective all aspects must be considered, the hardware, software, control, interaction, and educational purposes to which it is to be put. "The basic philosophy underlying the multimedia approach to instruction is that a number of different media (or channels) are used (simultaneously or in sequence) in order to implement a training or learning task. The computer acts as the overall controller of the resources and where possible selects those that are most appropriate to a particular instructional need [7]. One study that investigated the use of film, text and practice found that students retained more of the material when all three presentation methods were used [18]. Students were engaged in learning to construct a helicopter

from building blocks (akin to lego in New Zealand), where the final assessment was the students' ability to reconstruct the helicopter without assistance. Text was used as a reference only describing each type of block, the film and practice were used to show the actual construction tasks. A key conclusion from the study was that the correct mix of instructional methods/presentations for each student can yield enhanced results [18].

Targeting the audience and making the material and presentation relevant to them in terms of levels of interaction and complexity of the material itself can mean the success or failure of a multimedia presentation. When multimedia is used the success or failure has to date seemed more dramatic in comparison to other methods. For example, "As much as any other subject, engineers can use the multimedia approach to learning. As part of their commitment to support the profession through publishing, the Institution of Electrical Engineers has embarked on an exercise to explore multimedia. It has chosen to begin with the topic of Software Engineering. In a project funded partly by the European Commission, some four CD-ROM developments are being produced. The topics in this project were chosen carefully because of their appeal and their relevance. Although aimed initially at the continuing professional development market, it is anticipated that the materials will be of interest and relevance to undergraduates in the later years of their studies" [19].

One area in which multimedia has made an impact is the combination of various media to realistically express real world situations and scenarios. These are often typical situations for each topic and are by nature restrictive in their scope and interaction. Learner actions are often implemented as either right or wrong responses. The same basic strategy can be implemented as knowledge-based simulations, where a greater degree of subtlety can be built into user interactions. These systems utilise planning and/or user modelling to make assumptions about the situation and the user's state of knowledge and learning needs. With increased user response flexibility comes an increase in complexity of internal representations and the sophistication of control structures. Another component of such flexible systems is dynamic assessment, or on-line calculations and reasoning about user actions, which can provide real-time comparisons between student and expert actions, in the context of the situation. These systems are highly useable but require a development team of both educational, programming and task experts to produce efficiently with the required level of realism without neglecting the associated learning outcomes.

Developers must seek a balance between the evolving demands upon multimedia applications on the one hand and the evolving capabilities of multimedia on the other. Despite the volatility of multimedia programming, it is possible to identify some general problems confronting environment developers[15]. These include:

- Composition; how to structure and aggregate digital media
- Synchronisation; how to specify timing constraints and ensure their satisfaction during presentation
- Interaction; how to add interactivity to digital media
- Database integration; how to access large shared collections of digital media.

Further technological challenges for making multimedia tools more effective and widely available include:

- transitioning to knowledge-based multimedia systems that reason about the curriculum, student, and situation before responding [20]

- transitioning to network-based multimedia systems rather than individually equipped workstations [20]
- developing authoring tools to support rapid development of multimedia materials [20]
- increasing demand for location independent education with diverse delivery methods available
- the increasing amount of material being developed, requiring constant updating to remain relevant.

3.4 Hypermedia environments

Hypermedia can be defined as information in many formats that is linked together into a dynamically accessible presentation. The term is usually used, along with hypertext, to refer to material that is 'published' or presented in a manner suitable for distribution on the world wide web. The world wide web (WWW) refers to the structured assortment of information available on the internet. Hypermedia material can also be published in this manner to an intranet, ie within a local or organisational network, thus retaining control over its access and distribution.

A well constructed hypermedia systems can serve very well as an exploratory learning environment. Students are given freedom to explore a body of knowledge while engaged in a more directed task [21].

Without some form of direction most learning goals will never be achieved using hypermedia. Direction can come in many forms. One such form is structured browsing, where metaphors are used to assist and consequently restrict the accessing of information. One study found that "the most serious problems encountered in today's hypermedia systems are the well-known 'getting lost in hypermedia' phenomenon and the 'loss of homogeneity or incoherence of obtained information'. Both problems are in fact, problems of human-computer interaction and very closely connected with a so-called 'cognitive overhead' (ie an additional knowledge about structure and contents of a hypermedia database) needed by users in order to efficiently navigate through the hypermedia database. There exist methods of simplification of the hypermedia browsing by means of providing users with such a 'cognitive overhead' in a particular graphic form. The most well known methods are:

1. graphical representation of logical structure of the database (ie visualisation of a hyperspace)
2. using a variety of interface metaphors to assist users in navigating through hyperspace" [22].

Another study found similar problems for users of hypertext and hypermedia systems. The five principle areas of concern were "First, users get lost. Second, users may find it difficult to gain on overview of the material. Third, even if users know specific information is present, they may have difficulty in finding it. Fourth, learners may ramble through the knowledge on an unmotivated and instructionally inefficient fashion. Finally, coming to grips with the interface for controlling the various facilities may interfere with the primary task of exploration and learning about the materials." [21, 23]. The developers of the system used within the study quoted above have sought to address these issues by the provision of maps, indices

and tours that are integrated into the hypertext environment. Maps give a graphical overview of the structure of information at various levels giving an indication of the amount of detail carried within information presented at that level. Indices allow users to reference material dynamically and give links to all pages containing information on the topic sought. Tours allow the user to be guided through the material in a more structured fashion, presenting them with all the information on a selected topic in a logical sequence. "The study demonstrated that, even within the first 20 minutes, different access facilities can result in large differences in usage, and that facility usage is modulated by the user's task. Exploratory tasks resulted in greater use of tools for browsing the information; directed tasks resulted in greater use of tools supporting search. Taken as a whole, the results support our view that the way hypertext tools are used in search and in learning is highly task dependent: isolated studies provide small pieces of a somewhat complex cognitive jigsaw. Rather than attempt to understand this jigsaw, our practical approach has been to try to provide learners with a range of useful tools embedded within a simple interface, in the hope that the worst consequences of task dependency will be absorbed by the strategic options that the range of facilities makes available to the user" [21].

The issues raised by these studies identifies a number of challenges to the developer of hypermedia systems. In the second study mentioned above the use of guides in the process of navigating within the system formed a core to a solution to the problem of getting lost. The use of metaphors has proved to be no less successful in hypermedia environments than it has in other areas of human-computer interaction. The most prominent of these would be the desk-top metaphor used by both Macintosh and Windows based computers. This link with existing mental models also assists in the reduction of the cognitive overhead encountered in this environment. The ability to access information directly, and not to be restricted in the path taken to any specific piece of information is an important factor for the continued usefulness of the hypermedia environment once the initial goal has been attained. From a student's perspective the ability to annotate material and interact with it enhances the course of study and aids in the personalisation of the learning environment. These issues are reflected in the duality of challenges to the designer of hypermedia environments, especially instructional environments. The first set of considerations relate to the interaction with and navigation within the environment. The designer's challenge is to create tools that:

1. Are easy to learn, operate in a obvious way, and are easy to use.
2. Are integrated, appear seamless, and require a minimum number of steps to perform desired functions.
3. Reduce the cognitive load of students and instructors rather than increase it.
4. Reduce the difficulty of instructors generating materials to bring to the course and students copying materials to take from the course.
5. Promote active interaction with course materials and collaboration among the students and the instructor. [24]

The second set of considerations addresses the issues at a higher level. There is a certain amount of repetition between the sets. This should not be viewed as an overlap but more of a refocussing of these aspects to enable the issues as a whole to be addressed. These considerations are:

1. *Level of complexity.* Hypermedia databases add an additional level of complexity to the material. Prior knowledge about the material should

- facilitate the use of the interface, and conversely, use of the interface should facilitate learning of the material.
2. *Information granularity.* Often the point of instruction is to provide a look at the forest and not the trees. Consequently, the instructor must gauge the proper granularity of information and not allow the hypermedia to shift to another level at inappropriate times.
 3. *Information rate.* One can travel through hypermedia databases faster than one can process the information. It is important, therefore, that the instructor act as a guide and set a reasonable rate of information flow.
 4. *Extensiveness of the network.* The classroom provides the primary hypermedia database. In principle, the classroom could be extended to all electronic information. Here again, it may be important for the instructor to set limits on the domain of information accessible during the course.
 5. *Hypermedia and collaborative tools.* Success of the electronic classroom will depend on the interface tools that instructors and students have for browsing, authoring, and analysing the hypermedia database.
 6. *Price considerations.* Initial implementations are always extremely expensive. Funds are often overspent on facilities that may not be used and fall short on others that are under great demand. It is inconceivable that the electronic classroom of the future will not be so expensive, and, in fact, will be a natural outgrowth of the prevalence of hypermedia databases, local area networks, university networks, and personal computers. [24]

The exact implementation of any educational environment, hypermedia included, is highly dependent upon the domain or area of knowledge being addressed. It is all too easy for the designer and educator to start with the same goals in mind for the systems but what is actually produced can be conspicuously divergent from the original goals. The educator's chief task is for the students to obtain a preset level of understanding of the domain in question. "The task of understanding a domain is normally achieved more efficiently if learners can be persuaded to think actively about the structures and relationships within the domain. All too often, designers of CBL (computer based learning) conveniently forget that their goal is learner understanding rather than merely information provision. Additional demonstrations or mini-experiments may all help, depending on the nature of the domain." [23].

Many hypermedia environments are designed to be in place of, or supplementary to, existing educational delivery within the course. Hyperlecture [31] is an example of a system designed to be used both as a presentation aid within the standard lecture environment but also to be used later as the basis of the student's notes. The system can be used within the lecture to display material and as a prompt to the presenter. It is also able to display, on the presenter's system only, a graphical representation of the time taken to present each portion of the total material designated for this lecture. The students are able to annotate their copy of the presentation material with both the use of text and sound. Audio was added to the system specifically to address the needs of visually impaired students. Notes may be added by the tutor before release to the students, thus providing a fuller explanation and additional information while not cluttering the display with too much information. Notes could be added in a 'speech bubble' that appeared as an icon in the margin and when clicked with the mouse become an overlaid window displaying the additional material. This window disappeared again when the original window lying behind the note was clicked with the mouse.

While Hyperlecture employed the use of additional information to further explain specific points it was still static in terms of its implementation. Being able to generate explanations dynamically enables the provision of explanations that can take into account a greater number of factors such as the concepts the student has previously mastered, where the student is up to within the material, the level of assistance previously requested; thus providing increased relevance to the student. One particular implementation [25] of this approach utilises a combination of user modeling, adaptive explanation generation and hypertext/hypermedia techniques. Hypermedia can be used in a more flexible manner to construct explanations on demand and adapt these if the initial explanation is not fully understood. Referencing the user model enables the system to tailor an explanation to each specific user. It may encompass the use of terms the student has learnt previously. If the student has yet to master associated areas, concepts can be explained rather than just being referred to. Explanation can be simple at this point with the detail being in later sections. "This approach allows the system to generate menus of possible questions which seem to suit the working context and the situation that led to the need for an explanation, so that it reduces difficulties in user modelling and interpreting requests for further information. Hypertext follow-ups allow the user the freedom to escape from the system's assumptions about their information needs. They offer the user alternative information to explain the same point, and thus aid him or her in a better understanding." [25].

Taking the application of dynamically determining information presentation further has resulted in the development of intelligent learning environments. Each page, or segment of material, is constructed on demand to display specific information relevant to the educational tasks and model of the student's current level of understanding. The main features of such systems are:

- the hypermedia network is designed as the visualisation and externalised domain network
- an external representation of a knowledge element (a page) is generated or assembled from its internal representations (a frame)
- the context of the hypermedia page is adapted to the student knowledge reflected in the student model
- the links from any node to related nodes are marked visually, reflecting the current 'educational state' of the related nodes for the given student
- student interaction with the hypermedia component is reflected in the student model and can be used by other components of the intelligent learning environment [26].

The effectiveness of any hypermedia environment is related to core aspects, namely the educational goals of the system and the understanding and experience of the developer. Understanding of the 'construction' process itself is highlighted by one particular study which gave students, at university level, the task of developing a seminar paper in both paper or linear and hypertext or non-linear forms. Linear refers to the way in which the material is received, or read, by the reader. In the paper format the text is processed in a linear manner, from the start to the end on a single route. In a non-linear implementation, such as the hypertext version it is possible for the reader to construct their own route through the material, covering aspects in the order they choose rather than being locked into the way the author has presented it.

Students wrote the linear version first and then converted it to a hypertext format. Substantial improvements could be seen between editions of the hypertext form of the paper. Consistent 'errors' made by the students included

- first implementation was quite often linear
- no links were made to some pages unless accessed in a linear manner
- the amount of information on a page varied from too much to too little
- very limited use was made of audio facilities
- no graphics were included (although their use was encouraged) [13].

Like all things, good systems only come with the experience of producing systems. It must be accepted that the first few editions of any system are not going to be perfect, but continuous improvements must be made for the environment to mature and develop further.

3.5 Virtual reality and simulation

Modelling of real environments for educational purposes has been carried out for some time. The computer-based simulation of the operations of particular pieces of equipment or operations to be conducted in a restrictive environment are being increasingly employed. A well known example of this is the flight simulators used in pilot training. Looking inside 'closed processes' is another easily identifiable application of simulation, for example the reactions of various substances in a kiln. Some simulators have created a 'virtual world'. In this virtual reality the users interact with it as they would the real thing thus giving the illusion of reality. This illusion is best accomplished when both interaction and feedback have been simulated. For example moving your head changes the view portrayed, and the effects of pushing button or touching something is given as both visual and tactile responses.

"Virtual reality (VR) has excited the instructional technology community for a variety of reasons, including the notion that the instructional systems using VR interfaces are potentially more motivating than traditional two-dimensional interfaces. The VR interfaces allow learning to be situated in a virtual simulation of the tasks or information to be learned, possibly attenuating problems associated with inert knowledge. Inert knowledge is knowledge that learners fail to apply outside the classroom, even though it is relevant. VR systems may one day prove to be extremely cost effective interfaces for simulation-based training, since one VR system can simulate many target systems" [27]. For virtual reality systems to realise their perceived potential they must be created with increased levels of meaningfulness in terms of the presentation of information and the tools they simulate. The authenticity of the virtual environment should increase the ability of the learner to transfer knowledge and skills from the learning environment to the 'working' environment. It is not just the environment that is required to be meaningful but the instructional process as well. The more complex a task, and the more critical its correct operation, the greater are the benefits of VR and meaningful instructions.

To be implemented effectively the developers of virtual reality instructional systems must have knowledge, or access to knowledge of a wider range of subjects than most programmers. As one study found, "developers must have a thorough understanding of a range of diverse and often complex subjects, including the large-scale, dynamic

system, the task or function of the human operator in the system, artificial intelligence-based systems, on-line human-performance techniques, instructional strategies, and human-computer interface design” [28]. The level of integration of the different areas of expertise make the development of true virtual reality instructional environments a team based venture, especially if the resultant systems are to fulfil higher educational goals than merely creating an environment in which students are free to explore. Even the task of creating an environment or physical structure for VR is an extremely complex task.

Explorative learning, and the environments in which this is done have their place, yet should not be confused with intelligent instructional environments. Simulators used for explorative learning purposes are able to give students the ability to see aspects otherwise hidden from the observer and to encounter rare system conditions in a structured environment. Of importance to the educational provider is the ability of simulators to “allow students to explore new system configurations, discover new concepts, and effect new procedures at the students own pace and without damaging costly equipment or creating safety hazards” [28].

When adaptive instruction is coupled with the simulation the whole becomes an intelligent instructional virtual reality environment. It is possible to present instruction in the context of a simulated yet realistic situation. An investigative study [28] found additional benefits of simulator-based instructional programmes to include the following

1. A seemingly endless variety of problems, exercises, etc. can be generated.
2. The pace of the simulated process can be speeded up or slowed down to illustrate various concepts to the students.
3. Students can create and test hypotheses and models from the simulated data.

The same study records an interesting result. Subjects who ‘trained’ on a simulator performed poorer than subjects who also received on-line instruction coupled with the simulated environment, when tested using problems that had not previously been encountered. This reflects greatly on an educator’s desire to reduce inert knowledge in the student.

Interesting results are also being recorded where cognitive loading is under investigation, as applied to learning in a simulated environment. Simulated ‘real-life’ environments allow the learner to reduce the cognitive loading to a minimum. It enables learners to utilise more of the information stored in long-term memory leaving short-term memory relatively free to be used in the specifics of the current situation [1]. An example of this can be found in pilot training. Once a pilot has learned to fly, learning to land at various airports via a simulator is a process of adapting their existing general principles of how to land a plane to the specific requirements at this particular airport.

Although the relative benefits of utilising virtual environments in educational roles has been noted, there is still a cost-effectiveness decision to be made. The requirements in terms of hardware to support various virtual environments is often great and outside the financial restrictions of most educators. Although the cost of flight simulators for example is high, this is small compared to the cost of actual aeroplanes and the probable loss of human life in the worst case scenarios. The choice of hardware and software is dependent upon the environment being simulated and the degree of realism in presentation and interaction that is required.

3.6 Hardware, communications and support

There are many considerations and trade-offs to be made in the areas of hardware, communications and support. Often the purpose to which the hardware is to be put and the desire for it to be able to do multiple tasks dictates the types of hardware that are required. This in turn coupled with data transport factors will determine the type of communication used. The ease of use of the entire system and the targeted skill level of those who will use them contributes to the determination of the amount and type of support needed.

A computer is one of many hardware devices needed in a CME (computer mediated education) environment. The size and type of screen are dependant upon what is to be viewed. Are multiple screens to be employed? For interaction, is the learner to use a keyboard, mouse other pointing device, a touch screen, or a combination of these? [7] Irrespective of the type of display mechanism, it must be able to portray the data in the appropriate resolution to be used effectively. To this end most monitors available are able to display data at multiple resolutions, yet all will have a maximum capability. In the presentation environment there are also a number of projection devices with which information can be shown in a more formalised environment such as a lecture. These range from the LCD panels that reproduce the screen output of the PC and sit on top of a conventional overhead projector, to full standalone projectors. The standalone projectors, although most often still requiring a data source, do not need any other devices to be used in conjunction with themselves. The LCD panels require an additional light source, usually from an overhead projector which must be suitably powerful to project through the panel and onto a screen data that is readable by the audience. The size of room in which the presentation is to take place will often govern the choice of projection device. One device useful to the presenter is the Video Show Presenter by General Parametrics [29]. This unit plugs into the parallel port of any PC and gives a hand held display of whatever presentation is currently being executed on the computer. It also allows the presenter to move a greater distance from the originating PC by virtue of its cord length and still retain control of the presentation. The display and control features are built into the device which is small enough to fit comfortably in the presenter's hand in the same way as any remote control.

There are a number of circuit boards, or cards, that may be added to a computer to allow it to perform additional functions and/or enhance the performance of the machine in these tasks. When working with multiple media sources the capability to process this data quickly and in the various formats required becomes a critical issue. Examples of what is available in the area of video capture, compression and manipulation, include the Pro Movie Studio by Media Vision and the Smart Video Recorder by Intel [30]. Play-back is defined to be on a PC. Other cards like those available from AI Tech are able to receive broadcast television signals as well as the line inputs normally associated with video capture [30]. All of these cards come accompanied by software to assist the user in the capture, manipulation, storage and display of the video signals. In all instances this video will be accompanied by its associated audio as a mixed or composite signal. Separate cards are available for pure audio processing in a similar manner to the video just described.

With such a variety of data formats and companies manufacturing devices to deal with them, an internationally accepted standard is a necessity. "Standards are crucial to promoting multimedia - in the absence of standards developers are not users of stable platforms or accepted data formats and risk producing software and other products with little market value. Because of extensive commercial interest in the computer, communications and entertainment industries, there are many groups involved in standards and many sides to basic issues. For example: should standards focus on data formats, or also include presentation platforms? Is it too early to standardise, or would even the premature standards be a boon to multimedia development? Who should formulate standards, the traditional standards bodies or newly formed manufacturers and users groups? Multimedia standards should incorporate existing single-media standards and conventions, but go beyond these and address how different media can be combined and interrelated. Important issues include data representation, exchange of multimedia information, and presentation platforms" [15]. Examples of new standards can be taken from a range of data storage and manipulation available when stored on compact disc (CD).

- *CD-digital audio.* Refers to a 'standard' CD, one containing a digital audio recording. Compact disc plus graphics and compact disc plus midi are two early CD formats that store text (such as song titles) and either simple graphics or MIDI data in unused subcode.
- *CD-Interactive.* Provides a complete specification for the storage of multimedia data on a CD-ROM. In addition, CD-i specified the player hardware and software needed to extract, present, and interact with the data.
- *CD video.* Is a hybrid analog video/digital audio format. Three disc sizes are used: 12 inch allowing 60 minutes of video per side; 8 inch, allowing 20 minutes; and the standard 12 cm (4.75 inch) CD, called a CD-V single, allowing about 6 minutes of video. The 12 and 8 inch discs are usually referred to as laser discs or LD. In each case the video is accompanied by a digital audio (and possibly analog audio) sound track. [15]

The most commonly seen implementation is the CD-ROM (compact disc - Read Only Memory) which is a storage medium for arbitrary digital data, and as such can be used in all instances noted above. As the use of CDs to store data becomes more widespread there comes the issue of read/write computability between devices and the speed of access of these devices. Rapid advancement in technology has seen the drive speed for a CD-ROM move from 1x (its default listed speed) to 2x, to 6x and 8x and on to 10x. Speeds are still climbing causing some incompatibilities with discs produced for the lower speed or higher speed drives not functioning correctly within a device of a differing speed. The development and acceptance of standards needs to keep pace with the technological developments that each is addressing.

Independent of the number and type of multimedia devices used, control, communication, and synchronisation become critical to 'successful' system integration. Several options present themselves and come in two general categories. Dedicated hardware with associated control and presentation which is only used in a standalone mode of operation and computer controlled hardware that communicates directly with the multimedia device freeing the 'host' computer from possible computational overload in dealing with the specifics of every device and the corresponding 'explosion' in number of commands and size of the required operating system (OS). "It is not reasonable to build a lot of specific multimedia features into the OSs API (Application Programming Interface) because there are so many choices, and the field is growing rapidly. A better OS strategy is to create a multimedia API that is general, but can be configured ad hoc by installing proper

device specific drivers” [12]. This is the basic strategy adopted world wide. Applications utilise a standard syntax to communicate with a Media Control Interface (MCI) which in turn handles all interactions with specific hardware devices. The manufacturer of the device provides a driver or “interface with which the device communicates in a manner specific to itself with the MCI. The MCI has been “developed by Microsoft® with a consortium of software manufacturers” [12].

Developing systems that are capable of utilising a variety of interactive techniques and associated devices has also received research attention [31]. Such systems are capable of ‘presenting’ the same content independent of computer platform and device manufacturer over a distributed link. This is of specific interest to educators who work with learners who, for whatever reason, are separated in both geography and time.

As advancements are made in terms of both hardware and software a higher degree of ‘reality’ is able to be presented to the user. This can be seen in the use of a pointing device, usually a mouse, in conjunction with the more traditional keyboard for interaction with the system. There is also an increasing market for the utilisation of voice commands. The information presented to the user has likewise shifted from a purely text based display to one with a greater emphasis on graphics. Coupling these two with the availability of suitable communications bandwidth to transmit graphical information is reducing the physical separation of the parties involved. “In 20 years the overall telecommunications and technological environment will be utterly unlike anything existing today. The pioneers who developed group communications technologies in the 1970’s felt that its benefit could be quickly recognised and its mass adoption may only a few years away. Their confidence proved ill-founded, even through the 1980’s, and will probably remain so until the world had adjusted itself to the new political, economic, and technological changes in which it is now embroiled. It will be quite possibly the opening decades of the 21st century before computer mediated communication technologies are readily adopted by the international corporations of the next millennium” [15]. This prediction, published in 1991, shows a good understanding of the ‘extra’ factor in the development of new technologies. It also reveals the tendency for technologies to be developed for a limited number of reasons and then utilised for many others. When wireless communication and mobile computing are combined aspects of dynamically being able to address communication types and bandwidth along with user verification and file synchronisation becomes increasingly necessary. One such system performs these dynamic adaptations based on data type, location of the mobile client and the capacities of the communications mechanisms currently employed. In this particular system “the role of the operating system is to sense external events (such as connectivity and physical location changes), and to monitor and allocate resources (such as network bandwidth, cache space, battery power, and communication budget). In contrast the role of individual applications is to adapt to changing conditions by using the information and resources provided by the operating system. The nature of the adaptation is specific to the application. For example, in accessing video data, degrading the display from full motion colour to black and white, and thence to slow-scan might be appropriate. But when accessing map data, the meaningful form of degradation might be to lower the resolution and minimum feature size represented. The overall goal is to minimise the need for active intervention by users to cope with the consequences of mobility. For example, it is inappropriate to require the user to use a different video display programme when bandwidth drops” [26].

Once a system operation becomes distributed, for example over a network, the synchronisation of information passing and error handling becomes increasingly critical to a smooth seamless operation. "Clock errors, communication and processing delays make media synchronisation a very difficult problem in general. Design and evaluation of techniques for media stream synchronisation is an area of intense research. Some existing and proposed techniques include:

Interleaving. One way to simplify inter-stream synchronisation is by interleaving the storage of elements from several streams. In this way media elements are retrieved together and so at least start out in sync. This technique is the basis of CD-i layout and the 'audio/video interleave' (AVI) file format. As a side-effect of exploiting physical storage layout, interleaving 'hardwires' the timing relationships between streams. This is appropriate when presentation timing is known and unlikely to change, but ineffective if inter-stream timing can be modified or is not known prior to presentation.

Time code. Streams with embedded time code can act as references for other streams. Inter-stream synchronisation is then achieved by having components continually adjust their production and/or consumption relative to the time code, an ability known as 'time code chase'. Time code synchronisation also appears in software. An example is QuickTime where time code chase is realised by extending and dropping the display of video frames.

Synchronisation components. Consider a transformer component that encapsulates a 'timed' queue - elements arrive at the input port, are added to the queue, remain in the queue a certain time interval, and then are sent through the output port. This component is the digital version of a delay line, a device that shifts an analog signal backwards in time. Delay lines are used in audio and video studios to make fine adjustments in signal timing - by adding delays it is possible to ensure that comparable times are taken by signals arriving at the same point over alternate paths. In a similar fashion, a digital delay line, or 'delay component', delays one stream so it remains in sync with another stream suffering a comparable delay. Other forms of specialised 'synchronisation components' have been proposed. For example, the 'software phase-locked loop' uses a reference clock and a regulation process to adjust the rate at which elements are emitted from an output port. An average output rate is fed back to the regulation process to help ensure stability. It is also possible to create components that remove delay.

Load scaling. In situations where delays within components lead to timing errors, it may be possible to recover synchronisation by reducing component load. Consider a transformer which receives elements from an input port, performs some processing, and then outputs modified elements. If the component cannot satisfy the timing constraints on its output port it may ignore occasional input elements or simplify its processing task (and so perhaps reduce the quality of its output stream). In either case the effect is to scale down component load.

Stream managers. Several multimedia programming environments, noticeably AVK and MMPM/2 (IBM's multimedia presentation manager for OS/2) contain software components explicitly responsible for maintaining the flow of streams. Stream management is also used to control audio/video servers and file systems supporting time-based media. Stream managers offer user applications high-level interfaces to stream services, including, for instance, the ability to start, stop, pause, fast-forward, and cue streams. Internally a

stream manager may rely on interleaving, time code, some other synchronisation technique, or some combination of techniques. This is a choice made by the designers of the stream manager and is largely hidden from applications.

Intelligent scheduling. If delays can be anticipated then one can ensure that streams arrive in sync by careful scheduling of source start-up times. [15]

Synchronisation issues also occur when multiple media formats are to be presented simultaneously. "Future advances in networking will make it feasible for computer networks to support digital multimedia transmission. Coupled with the rapid advances in storage technologies, they can be used to build multimedia on-demand services over metropolitan-area networks (such as B-ISDN) that are expected to permeate residential and commercial premises in a manner similar to existing cable TV and telephone networks. A critical requirement in building a multimedia on-demand service is the need to guaranteeing continuous playback of media streams (since media quanta, such as video frames or audio samples, convey meaning only when presented continuously in time, unlike text in which spatial continuity is sufficient). In order to generate continuous playback, (1) the multimedia server must support continuous retrieval from a disk, (2) the network subsystem must guarantee timely delivery of media quanta to the display sites, and (3) the display sites must avoid buffer overruns or starvations." [16]. When this is achieved video and the associated audio segments are retrieved and played at the same time resulting in no perceived delay by the user.

Considerations must also be made to the servicing of multiple clients. Each client requires the same continuous playback of requested media segments. This could be the same sequence at the same or differing times, or of different sequences. The policies of servicing multiple clients can be classified into two main categories: deadline based and placement based. Whereas the former retrieves media blocks based on the earliest deadline first scheduling policy, the latter retrieves media blocks from disk to minimise the total seek and rotational latency (total time to locate and retrieve data from disk) incurred during retrieval. Servicing policies can be applied either to the media blocks within a strand (yielding a local schedule) or the global pool of medial blocks form all the strands (yielding a global schedule). Clearly, when servicing policies are applied among media blocks within a stand, the multimedia server has to employ ordering techniques (such as round robin ordering) to switch from one strand to next during each round. "In the Quality Proportional Multi-client Servicing (QPMS) algorithm, the number of blocks accessed during each round for each strand is proportional to its playback rate. Since the display of media strands proceeds in terms of quanta such as frames, retrieval of a fraction of a frame cannot be used for display, causing the display to starve until the remaining fraction arrives, possibly in the next service round. The data transfer requirement of a rope (many strands) may vary during its playback. Given the relationship between the playback intervals of media strands, the data transfer requirements of a multimedia rope can be completely characterised by maintaining: (1) the time instants at which the playback of its constituent media strands begin and end during the playback of the rope, and (2) the extent of increase or decrease in the data transfer requirements." [16].

In reference to total systems integration and application, support and environmental issues must also be considered. Environmental considerations involve issues of adequate room size for style of presentation in conjunction with total class size,

lighting to ensure that everyone can see adequately when viewing presentations while retaining sufficient illumination to take notes, appropriate screen size for both the room and types of data being presented. Added to these is the consideration of portability. Is the system to be used in a set location and left as a permanent fixture and/or is it to be made transportable to various locations. If the system is to be portable, equipment needs to be smaller and lighter than is necessary for a fixed installation. There is also a question of how many people are necessary in terms of transportation and setup/pack up of the equipment. Are support personnel available at all locations at which the system is to be used, and if not are they contactable by other means (eg EMAIL, phone). If the delivery situation is a presentation in a form more akin to a lecture, and especially if the audience is at more than one location, adequacy of display equipment is essential. The system must be able to portray the desired information at the right resolution and at an acceptable speed. It must be remembered that a "slow performance equals a bored audience" [17], and in terms of educational settings a bored audience is not learning anything.

Support systems are made up of many aspects. Assistance in terms of technical expertise can be called upon when it is needed whether or not the person concerned is employed by the same organisation or on-call from an outside company. Educational guidance is needed to assist learners in the tasks which they must complete, to ensure that cognitive overload is avoided as much as possible, and to provide additional assistance on an individual level for each learner as they require it. It should always be remembered that for some students the use of a computer can vastly increase their cognitive load. For others who are competent and confident the load can be effectively reduced [Chapter 1]. Any computer based systems will, by its nature, require written documentation to accompany it. This should be written and designed to fulfil a specific purpose not just done so that it can be said that it exists. Technical reference material will be needed by those supporting the systems, operational reference material is needed by both educators and learners alike, navigational material is often usefully provided in a quick reference guide that can be issued to learners or kept with the systems itself. Good documentation should never be underestimated, and should contain information that is easy to find, contains the correct level of information for the target audience, and make good use of graphics. [17] The use of graphics taken directly from the applications in operation can relieve the stress of a learner when faced with a written description that uses language they do not yet fully understand. It can also reassure the user that they have successfully executed the operations to reach the point in the task execution depicted.

3.7 Concluding comments

The application of technology within education must always be done appropriately. Many considerations are required, from both educational and technical perspectives. The final selections will be governed by the material itself, geographical location of students, special needs of student and course, the methods of delivery, maintenance issues and finance. All systems must operate within limitations and the final implementation of any system will reflect this. All possible technologies and the application of such technologies should be fully investigated before a decision is made. The over-riding factor in technological solutions within education is

appropriateness. The technology should enhance and not detract from learning, whatever the topic or situation.

3.8 References

1. John Sweller
Some Cognitive Processes and their Consequences for the Organisation and Presentation of Information
Australian Journal of Psychology, vol 45, no 1, April 1993
2. C Frasson, M Kaltenbach, J Gecsei, J-Y Djamen
An Iconic Intention-Driven ITS Environment
Intelligent Tutoring Systems, Second International Conference, ITS '93, Montreal, Canada, June 1992 - Proceedings
3. P Robert-Jan Simons, Frank P C M DeJong
Self-regulation and Computer-sided Instruction
Applied Psychology: An International Review, vol 41, no 4, 1992
4. Kenneth Silverman, Ogden R Lindsley, Kathy L Porter
Overt Responding in Computer-Bases Training
Current Psychology: Research & Reviews, Winter 1990-91, vol 9, no 4
5. Jeff W Rickel
Intelligent Computer-Aided Instruction: A Survey Organised Around Systems Components
IEEE Transactions on Systems, Man and Cybernetics, Vol 19, No 1, Jan/Feb 1989
6. Rainer Gotze, Helmut Eirund, Roland ClaaBen
Object-Oriented Dialog Control for Multimedia User Interfaces
Human-Computer Interaction, Vienna Conference, VCHCI '93, Fin de Siecle, Vienna, Austria, September 20-22, 1993
7. Philip Barker
Author Languages for CAL
Macmillan Education Ltd, 1987
8. Friedrich Strauss
Context sensitive Help-facilities in GUIs through Situations
Human-Computer Interaction, Vienna Conference, VCHCI '93, Fin de Siecle, Vienna, Austria, September 20-22, 1993
9. M Satyanarayanan, B Noble, P Kkumar, H Price
Application-Aware Adoption for Mobile Computing
Carnegie Mellon University, School of Computer Science, CMU-CS-94-183
10. Peter L Brasilovsky
A Framework for Intelligent Knowledge Sequencing and Task Sequencing
Intelligent Tutoring Systems, Second International Conference, ITS '93, Montreal, Canada, June 1992 - Proceedings

11. Kaica Vaanoen
ShareMe: A Metaphor-based Authoring Tool for Multimedia Environments
Human-Computer Interaction, Vienna Conference, VCHCI '93, Fin de Siecle, Vienna, Austria, September 20-22, 1993
12. Arch C Luther
Authoring interactive multimedia
AP Professional, 1994
13. Margit Poht, Peter Purgatheofer
The Problems of Untrained Authors Creating Hypertext Documents
Human-Computer Interaction, Vienna Conference, VCHCI '93, Fin de Siecle, Vienna, Austria, September 20-22, 1993
14. H E J Borst Pauwels, J Sousa Pinto, B Sousa Santos, J A Martins
An Integrated Courseware Editor Based on OLE Technology
Human-Computer Interaction, Vienna Conference, VCHCI '93, Fin de Siecle, Vienna, Austria, September 20-22, 1993
15. Simp J Gibbs and Dionysios C Tsihrizis
Multimedia Programming
Addison-Wesley Publishing Company, New York, 1995
16. P Venkat Rangan
Architectures and Algorithms for Digital Multimedia On-Demand Servers
Performance Evaluation of Computer Systems, Joint Tutorial papers of Performance '93 and Sigmetrics '93, Springer-Verlag, Berlin, Heidelberg, 1993
17. Joe Hutsko, Les Howles, Connie Pettentigil
Room with a view
PC World, October 1993
18. Patricia Baggett
Learning a Procedure from Multimedia Instructions: the Effects of Film and Practice
Applied Cognitive Psychology, vol 1, 1987
19. Andrew McGettrick
Multimedia for Engineers
IEE Software for Engineers Supplement, March 1994
20. B P Woolf, W Hall
Multimedia Pedagogues, Interactive Systems for Teaching and Learning
IEEE Computer, May 1995
21. Nick Hammond, Lesley Allinson
Extending Hypertext for Learning: An Investigation of Access and Guidance Tools
Department of Psychology, University of York, England
22. H Mourer, F Kappe, N Scherbakov, P Srinivasan
Structured Browsing of Hypermedia Databases

-
- Human-Computer Interaction, Vienna Conference, VCHCI '93, Fin de Siecle,
Vienna, Austria, September 20-22, 1993
23. Nick Hammond
Teaching with Hypermedia: Problems and Prospects
Department of Psychology, University of York, England
24. Kent L Norman
*The Electronic Teaching Theater: Interactive Hypermedia and Mental Models of
the Classroom*
Current Psychology: Research & Reviews, Summer 1990, vol 9, no 2
25. Fahri Yetim
User-Adapted Hypertext Explanations
Human-Computer Interaction, Vienna Conference, VCHCI '93, Fin de Siecle,
Vienna, Austria, September 20-22, 1993
26. Peter Brusilovsky, Leonid Pesin, Mikhail Zyryanov
*Towards an Adaptive Hypermedia Component for an Intelligent Learning
Environment*
Human-Computer Interaction, Third International Conference, EWHCI '93,
Moscow, Russia, August 1993
27. J Wesley Regian, Wayne L Shebilske, J H John M Monk
Virtual Reality: An Instructional Medium for Visual-Spatial Tasks
Journal of Communication, Autumn 1992, vol 42, no 4
28. Janet L Fath, Christine M Mitchell, T Govindara
An ICAI Architecture for Troubleshooting in Complex, Dynamic Systems
IEEE Transactions on Systems, Man and Cybernetics, vol 20, no 30, 1990
29. Paul Worthington
Power Tools: Multimedia products for creative users
Personal Computer World, October 1993
30. *Short reviews: Upgrade, Multimedia Authoring*
Personal Computer World, September 1993
31. Damina Conway
HyperLecture: A Self-organising Lecture Presentation and Review System
Human-Computer Interaction, Third International Conference, EWHCI '93,
Moscow, Russia, August 1993

4 Programme Development

In any course a balance must be reached between established educational theory, the material itself, the designated function of the course (introductory or advanced, short course or full-time programme), and physical location of all involved (students, tutors, equipment). Development of an educational programme often falls to the person who is responsible for the delivery of the course. This person is not always fully conversant with relevant educational theory, but more often knowledgeable in the subject matter concerned.

When developing a course that is to utilise Computer mediated education (CME) it must be borne in mind that educational goals should not change just because of the introduction of computer mediation. The methods of delivery and assessment will alter to make the most of each method that is considered appropriate. Several major sections will need to be addressed: these are course design, topic selection, delivery mechanisms to be used for the material, related security of student work, student assessment and course evaluation.

Whatever is forcing the changes in the education system the process used to alter the system remains remarkably constant. This chapter presupposes that the decision to change the way teaching is undertaken has been made in an educated and reasoned manner and that computer mediated educational systems are appropriate to the course. The current situation within the Department of Production Technology, in which this thesis is based, a number of governing criteria have been decided upon. The primary objective is to establish a more flexible education programme for the Bachelor of Technology degree options taught by the Department of Production Technology.

The external factors contributing to this decision are the expansion of the university into a multi-campus environment, and the ever increasing number of students enrolling for the Technology Degree. The rise in numbers is taking place at each site individually thus increasing the staff to student ratio significantly. In some courses this is now 1 to 250. This is then multiplied by the number of campuses on which the course is taught, although not all campuses have the same size classes, and the ratio can rise to figures in the order of 1 to 400. These levels make it exceedingly difficult to continue to provide the levels of staff contact each student has enjoyed in the past, and for staff to keep track of individual students. Internal factors include the decision by the university as a whole, and its implications at departmental level, to provide a quality tertiary level education for all students. This has led to the desire to provide students with the opportunity to learn in a way best suited to their individual needs and styles of learning.

With students being able to select their best methods of instruction, resulting in an increase in their individual responsibility for their own learning, not all are partaking in every aspect of the course delivery. This relieves the tutor of dealing with the students that have grasped an issue to spend more time with those who are having difficulty thus making better use of both staff and student time. These developments are being introduced into the undergraduate programmes, yet are equally applicable to post-graduate level studies. Of special interest in this regard is the establishment of an in-situ Master of Technology programme, where students remain in employment while undertaking further study. The same techniques can be used for

specific short courses enabling technologists to partake in continuous professional development upon graduation.

4.1 Course design

When seeking to introduce more flexibility into a course of study a total redesign of the course structure is required. To enable the production and verification of multiple formats and delivery mechanisms, we must start the development process much earlier. It is no longer possible to alter the topics to be covered part way through the course. If students are to have access to the entire course material for the duration of the course then its content needs to be set and not altered for the duration of the course. To give students the flexibility to progress at a speed applicable to themselves, yet still within the total course time constraints, material cannot be released on a time dependent basis.

In a more flexible approach to education the roles of tutor and student are redefined. No longer is the tutor the source of all knowledge and the student a passive recipient. With increased flexibility comes increased choice. This choice is in terms of how each student will learn the material and gain the required skills, and in the degree of active participation they have in the direction of their own education. As the students' role changes so does that of the tutor. The image of a coach or director in the educational process is more appropriate in this new environment.

When a course is being redesigned with flexibility in mind several alternative mediums of delivery must be considered. Yet for each, the material must be well designed and encompass good instructional theory and techniques. The material and the medium of delivery must match each other for maximum effect and benefit in the student's learning. A well designed and structured course consisting of subject or topic units, and even sub-units, enables both student and tutor to gain a measure for the complexity and depth of the course. It also enables assessments to be set at known points, that can be taken whenever the student reaches them. This modularisation of courses allows more flexibility for the course designer in that a course may be altered by updating or exchanging modules independently causing only minimal disruption to the entire course. Once a group of students have embarked upon a course of study it should not be changed until after their completion, or before the next group starts, if this is before the previous group of students have completed the course. Students should be able to complete the same course that they started, especially as this relates to individual topics or subjects.

Ensuring that course material is available throughout the course gives students the opportunity to alter the pace at which they study. Some find a constant amount of time each week is best, while others wish to shorten or extend both individual sections and the entire course. Presenting the material in more than one format allows for a greater flexibility in how students access the course content. It enables them to put to good use the identification of their learning styles and to utilise many styles to reinforce each other giving a more rounded approach to learning. The different methods of delivery of material also enables students to study in an environment that suits them, be it at home, in a lecture or small study group environment, at a place of employment (especially for the in-situ Masterate programme), or at another location, even another educational institution.

4.2 Topic selection

Increasing the flexibility of an educational programme involves a number of steps. The indicators of flexibility that were chosen are that the students have more control over where they study, when they study, how they study and what they study. All this is by necessity kept within the overall university year and paper structure. To achieve this a number of didactic strategies are used to present the student with both choice in delivery mechanism and material representation. Each strategy has its own characteristics. When selecting the topic to be addressed the different strategies are placed alongside the subject material to give a comparison in teaching methodologies that can be used and how each one relates to different topics. No one strategy is capable of covering every topic or learning style of the students, yet with the use of a number of implementations of several strategies coverage of this diversity can be attempted. The didactic strategies under consideration are: human tutoring, computer-based tutoring, explorative learning, simulation, intelligent tutoring, and collaborative learning.

When implementing an educational programme in a flexible manner it is important that the structure and duration of each course is clearly stated and strictly adhered to throughout the course especially that which relates to assessment due dates. The execution of some aspects of a course may be determined by their own nature, particularly in relation to practical components. Requirements for laboratory exercises, and the timely execution of these, is dependent upon the type of course and manner of execution. Some of these location dependent components of a course will restrict the progress of faster students. This is offset and balanced with the associated importance of group work and team development. If team work is a part of the course then the associated constraints apply. These should all be set out and explained to the students at the commencement of the course.

It is possible, and indeed imperative for flexibility that several didactic strategies be used to present the same material. This allows the student to utilise one or more presentation methodologies to best achieve their educational goals and an understanding of the material. With this increased flexibility comes increased responsibility on the part of the student for their own learning. They must be aware of the best ways in which they learn; which presentation media, speed of presentation, the amount of practice or application examples that is required, etc.

Coupled with the choice of strategy is the wise selection of presentation mechanism. Each topic will have associated with it characteristics that should be used when deciding upon presentation formats. For example it is much easier and more sensible to use audio when discussing aspects of musical performance or simulation when explaining the change glass undergoes when placed in a furnace. This does not exclude the use of any other presentation mechanism, yet for best results material and presentation must be matched.

In this thesis we are concerned about the use of computers in the educational process and it is important to remember that any didactic strategy, and the various technologies that can be used within them, are only tools and should be treated as such. The computer is capable of more than static presentation of information and can assist in all of the didactic strategies mentioned above. Table 4.1 below shows each strategy and some of the possible uses for a computer within them. (This is not an exhaustive list.) The storage and maintenance of student records is something that

can be done on a computer independent, of the didactic strategy being used and so is not included in the table.

Strategy	Examples of uses
Human tutoring	<ul style="list-style-type: none"> • Generation of presentation material • Presentation of material in electronic form (eg onto a screen, sound clips, use of material stored on CD ROM) in the context of a lecture
Computer-based Tutoring	<ul style="list-style-type: none"> • Guidance to student on topic order (usually fixed) • Enables individual presentations • Increased availability of repeated presentation • Assessment of student knowledge • Practice exercises
Explorative Learning	<ul style="list-style-type: none"> • Unrestricted movement through subject material • Increased student control • Increased “excitement” factor • More informal way of learning • Encompass the use of multimedia effects to enhance presentation, “make it more real”
Simulation	<ul style="list-style-type: none"> • Presentation of machines or environments in a safe manner (non-hazardous manner, eg the inside of a furnace, a volcano erupting) • Allows students to make mistakes where the results don’t endanger anything or anyone • Enables practice of tasks to be carried out before “doing it for real”, eg surgery
Intelligent Tutoring	<ul style="list-style-type: none"> • Assists students on an individual basis • Alter the presentation and course through the material based on a model of the user • Increased avenues for tracking student progress • Can make use of student model when selecting an presenting assessment tasks
Collaborative Learning	<ul style="list-style-type: none"> • Asynchronous communication within and between groups/tutor • Collection and presentation of group

	generated material • Group based administration tasks
--	--

Table 4.1

All these changes in the way courses are taught must also fit into the established degree structure and mission of the department in which it is implemented. Major changes in the underlying ethos of courses can cause undue resistance by staff which is often reflected in the way they approach change in 'their courses'. Ownership of a course by those who are going to teach it is a significant factor in the unwritten attitudes expressed by the tutors to the students, along with enthusiasm for the subject itself.

Any alterations in course delivery are going to be reviewed critically by the students and institutional management upon the completion of the first examinable unit. If the students do not understand or refuse to alter their mental model of the delivery of education then the unit will not get a favourable response. A certain amount of flexibility needs to be allowed for in the description of the course for the introduction of the different methodologies that are to be used, especially those involving an assessment that is to go towards a student's final grade. Where an implementation method has proved successful it should be retained and improved upon. Investigation can be carried out to determine what aspects contributed to its success, with a view to the implementation of something similar within another course if it is deemed appropriate. With a view to using the knowledge and experience gained in courses in which these practices have been used in other courses which could benefit from it. There is also an opportunity to expand the generalisation of knowledge and experience into the curriculum as a whole.

4.3 Delivery mechanisms

Within each didactic strategy implementation there are a number of delivery mechanisms available. Selection of these is as important as the selection of the strategy itself. Material can be split into two groups in terms of a time based delivery. These are asynchronous and synchronous methods. These will be dealt with separately. The degree to which students utilise each method will be determined to suit their individual needs, often defined by their preferred method of learning and the topic concerned. Both methods are normally implemented simultaneously and used in a non-exclusive manner. An existing example of this mixture are students who undertake extramural studies. The bulk of their study is done in an asynchronous mode. Short residential courses, often one week in duration, provide the synchronous parts of the course, for example laboratory work and face-to-face group interaction.

4.3.1 Asynchronous delivery

Asynchronous communication is where the two (or more) parties are not present at the same instant of time. This is analogous to message passing where each person adds to an existing message and passes it on to the other, or next, person. As this delivery method is not time dependent it offers a greater flexibility in the way students access the study material. It also places greater constraints on the tutor. The material must be accurate and needs to be prepared well ahead of the time the student needs to use the material. When material is prepared for an extramural course it must be ready, printed, and sent to each student so that they may start the course at

the prescribed time. Also it is more difficult for students to query the lecturer on possible typographical errors that may have crept into the material.

When computers are used as the 'delivery agents' the method in which material is presented may take on a more dynamic form. Printed material, although useful, remains static and can add confusion to a topic where a simple animation sequence will show inter-relationships between and movement of component items more easily. Computer delivery may be in the form of a prepared disk or CD-ROM sent out to students or the utilisation of the dial-in facility of the university network. In some cases it is possible to use the network to control the passage through material on disk or CD-ROM.

The network can be used primarily in two ways, firstly the student may dial-in using a modem over the telephone lines and log-on to the network in the same way as a student in an on-site computer laboratory. Secondly the student may access material via the world wide web. This is often the preferred choice as the student can obtain web access from a local provider reducing the need for expensive toll calls, and they do not enter into the network itself. Web access to material can be password protected, coupled with a valid registration, thus providing a certain level of security for the educational provider as to access to the material, and for the student when submitting assessable material. It is important that only students that are currently enrolled in a paper may access the course material for that paper, or to have access to the assessment material. From the educational providers perspective this is for financial and educational copyright reasons. From the students perspective access also has financial ramifications as they pay for each course undertaken, and a desire to complete all sections of the course to the best of their ability without the interference of other persons not a part of the course. The security of data is dealt with in a later section of this chapter.

Interaction facilities, for example chat rooms, can be provided which gives students an easily accessible communication mechanism with each other. Some areas, or rooms, may be accessible to only a small group of students working together on a project, or to the whole class but not the tutor.

Examples of delivery modes available in a computer, or electronically based, environment include:

- Notes
 - Developed by the tutor or taken from texts on the subject offering greater degrees of explanation.
- Simulation sequences
 - These fall into two categories. Firstly, the interactive simulation of a procedure that is developed to run on the student's own computers. This would most likely be downloaded in the form of an executable file. Secondly, the ability to interact with 'real' equipment via a live link to the laboratory at the university or other institution. This can be done using the World Wide Web and specifically designed programme units, or via application sharing facilities.
- Animation
 - This differs from simulation where everything is represented exactly as it would in the live situation. Using animation we can take the student "inside" the process. An example of this would be the representation of glass inside a furnace or moulding machine or the forming of chemical

compounds at the atomic level. These portions cannot be seen in the normal observations of the process.

- Student queries of the tutor or fellow students
Commonly achieved via electronic mail. Students have time to formulate their question, asking exactly what they need to. Tutors also have the time to reply in an informed manner. If information needs to be collected to assist in the reply then it can be given and consistent replies made to all students. This mechanism is also useful for students to communicate with each other. It represents classmates assisting each other's understanding of the material being covered.
- Open forum discussions
Good use can be made of electronic news groups for this purpose. They are unmoderated and usually not well patronised by tutors. When used within an educational setting they can be more formally structured to specific uses, but their general governance still remain.

4.3.2 Synchronous delivery

Synchronous delivery is where the tutor and student are present at the same instant of time. It does not mean that they are required to be in the same location. Audio and video conferencing techniques make it possible for tutor and students to be in geographically distributed locations yet still maintaining synchronous communication. The options available under this mode of delivery are:

- Lecture
This is the standard format for the delivery of university course material, where the tutor stands at the front of a group of students and speaks on a topic. This can be with or without visual aids to both parties. It tends to be one way in terms of information flow. This mode is usually used when a portion of information is to be presented to a large number of students simultaneously.
- Tutorial
The tutorial is more interactive than a lecture and usually operates with a smaller number of students. Topics are reduced so as to be able to accommodate a greater participation by the student within the constraints of allotted class time. This is often used as a question and answer session.
- Laboratory
As with the tutorial this is undertaken with smaller groups, often with many groups performing experiments concurrently. Laboratory work is included within a course to demonstrate the application of what is being studied, and so that students may gain the necessary practical skills to implement their knowledge in later employment.
- Demonstration
Used when a practical representation is required and often, because of time constraints, it is not possible for the entire student population to perform the tasks themselves. Other considerations are the location and equipment required. Demonstrations can also be recorded and used in the asynchronous mode as a refresher aid at a later time.

4.4 Student assessment

The assessment of learnt material has several functions. For the student these include; an indication of their level of understanding and competence, identification

of areas that need further work, and a final grade. Each aspect is important and emphasis is placed upon it at different stages of the learning process. For the tutor a number of other factors make student assessment important. In conjunction with the need to track each student's progress through the material, tutors also wish to assess the effectiveness of the teaching methodologies used, and to obtain feedback on areas of refinement and improvement of the course often while the course is underway rather than waiting for the results of examinations and assignments at the end of the course. This is covered in a later section on course evaluation.

The type of assessment to be used is chosen to match the desired aspects of learning to be measured. The style of implementation should also relate to the mechanisms used in the learning process, for example practical skills should be examined in a practical manner. In a university setting the traditional form of assessment has been in written or occasionally oral examinations. Increasing use of smaller modularised units of learning has enabled the implementation of more specific and topic driven assessment methods. Four of the most commonly used assessment mechanisms are examined below.

4.4.1 Tutorial examples

Tutorial and self paced learning modules need to include a tracking mechanism to ascertain the progress of each student through the material. This information is used by the student and tutor for much the same purpose. "How far have I got in this unit?" Short question and answer sections at the end of each unit can be used to test the understanding the student has gained of the material and to point out areas that require further work. They can also indicate any area of the course that is being seen as difficult or ambiguous by a number of students. If this is so, then appropriate action can be taken by the tutor to clarify the issue and to determine what improvements should be made to the material.

4.4.2 Self assessment modules

Self assessment modules in addition to those in the tutorial itself are used to track the student's understanding in a different setting. Questions contain less precise information and are centred on problem solving rather than on a narrowly defined function or operation. This is of the same benefit to the student as tutorial questions in the standard university course. They may or may not be used by the tutor for formal assessment purposes. Students should be able to complete these assessment modules at any time and as many times as they like. This enables them to progress at their own rate and to repeat sections or whole modules if necessary.

With this modular format for self assessment units it is possible for the assessments to be made available on a computer, allowing the student to repeat sections as they desire. It is possible to construct questions, submit answers for marking, have the unit marked, and returned completely by computer and utilising the computer network. Students are given access to all available tools to assist in the completion of the questions. The ability to send a file via a computer network to another computer which recognises it, executes a programme to mark the answers submitted, and returns the marked file plus comments to the student enables a faster turn around time. The speed of turn around of an assessment unit is critical for the student to determine areas which need improvement and to decide on the appropriate action to be taken next. It may be possible to build in to the marking programme a mechanism

for determining areas of improvement, but at the very least it will be able to send notification of student results to the tutor who is also able to assess the situation and take appropriate action.

4.4.3 Assignments

Assignments are the first step in the formal assessment process. The marks gained on these contribute to the final grade for the course. As such they are taken more seriously by both the student and the tutor. More checking to ensure correctness is required and that the execution of the assignment is done within the course regulations in terms of the number of attempts each student is able to have and the time frame within which it must be completed. The same computer and network based mechanism used in the tutorial modules can be used here. This enables the students to become familiar with the mechanism for assignment completion and submission before they attempt an assignment that is to contribute to their final grade. The prior use of the system ensures that the student is not faced with two levels of assessment, the material and the system, at once. The system of completion should be transparent to the student at the point at which they attempt the assignment. Further discussion on the construction of assignments can be found in the following chapter. Good written instructions as to the execution of the systems and the submission of students' responses must accompany the assignment questions.

4.4.4 Formal examinations

Formal examinations pose another problem for the tutor. To enable the course of study to be more flexible the final examination must also be accessible by the student at any time throughout the duration of the course. Institutional constraints placed on the duration and delivery of courses places time and facility limitations on the execution of examinations. They must be completed by a certain date and be invigilated by an independent person or persons.

If the examination is to be completed in the same manner as the rest of the course was assessed and computers were used in the execution of this assessment, then the use of computer rooms for the examination becomes essential. This may cause additional scheduling problems with large classes, yet the availability of the final examination throughout the course duration will mean that not every student is sitting at the same time. A final session at which the examination can be taken is needed to mark the formal end of the course. As the examination is available throughout the course its construction must be such that no advantage can be gained from having seen it before. If students are permitted to sit the examination more than once it is required to be robust enough so that understanding of the material is being assessed not a student's ability to memorise examination questions.

4.5 Data security

As course sizes enlarge it becomes increasingly difficult to ensure the security of information passing between tutor and student, and between students. This is more critical in terms of assessment material and examinations than the delivery of the course content. Inter-student communication becomes important again due to assessment considerations and group interactions. Collaboration between students should be encouraged but not collusion and cheating.

The responsibility for the security of information lies with all concerned. Students must make sure that they are not willingly and knowingly cheating, this mostly comes down to personal integrity. This is important as the student may be punished in the same manner as the one cheating or required to redo the section or assessment concerned. Tutors and the educational institution have a responsibility to ensure that critical information is passed in a secure manner. The institution has the responsibility of securing the communications facilities used, and the tutor for mostly office security and information disclosure. No communications mechanism is ever totally secure but constant serious endeavours must be made to ensure they are as secure as possible. As education is ostensibly a contract between the student and the institution represented by the tutorial and administrative staff, each party to the contract has certain responsibilities and should be accountable for them.

As information is increasingly being passed via computer networks, the security of these is of greater concern to those wishing to use them for the movement of critical information. To students, and the educational institution, information that is to be used for assessment purposes falls into this critical category. The most common ways to help secure electronic access and transmission is encryption and validation (usually passwords) techniques. Without the appropriate “key” access is either denied and/or information unintelligible if intercepted.

4.6 Course evaluation

To monitor the way the course is progressing, being received and used by the students is as important an assessment to make as that of student progress. Part of the flexibility that the student enjoys is that the course is adaptable to the changing needs of all students. Some individual tailoring is possible but not every variation is viable for the tutor to be able to offer in any one course, without the assistance of specialist staff.

Course evaluation involves determining areas of improvement in the way information is conveyed to the students, how and when it is assessed, along with the performance of the tutor. Tutor performance is a widely debated topic in educational circles. A balance needs to be taken between the perspective of the student as a recipient/participant and the role of the tutor as a coach/participant in the educational process. Part of evaluating the role of the tutor is to accommodate the change in focus for the tutor from the provider of all knowledge to the director of investigation, the coach in the educational process.

Student participation in the evaluation process reinforces their active involvement in the way they are taught. It can increase the sense of responsibility the student feels for their own learning, and their ability to look critically at aspects of their education. This sense of involvement can be increased by incremental evaluation of the course throughout its duration as well as at the completion of the course.

5 Programme Implementation, Evaluation and Refinement

Changing a traditionally taught course of study into one presented and administered in a more flexible computer mediated environment is no small task. As outlined in Chapter 4 it requires reassessing the entire course with regard to content, delivery methods, timing, compulsory components if any in the course, and assessment. For this reason it was decided to implement pilot projects representing as many of these areas as was possible. These have covered the delivery of both interactive tutorial type information and the purely informative delivery of instructions, along with an almost complete course (student) assessment section. Thus it has been possible to implement as much as possible within 'live' courses while not creating too much disturbance to the general running of the course.

It was the intention at the outset of the process of implementation to minimise the adverse impact on student education and so only one aspect of development was undertaken in any one course. Several trial implementations have been produced that were evaluated before inclusion within any one course. Progressive refinement has enabled continual improvement of each implementation. Student feedback has been a valuable tool in this regard, and the assistance given by several classes is appreciated. The systems represented here reflect many of the stages of the development process. Each system is at a different point of refinement and included as such to demonstrate the development cycles. The conference information system is complete to two revisions and approved for use by the conference organisers in its current state. Further refinements can be made and implemented in subsequent years. The tutorial systems show two levels of refinement. The first is as it was when tested for the second time by a selected group of students having completed the course the previous year. These students evaluated the systems for useability, clarity of presentation, and usefulness within the designated course. The second tutorial system shows the inclusion of these comments and the further refinement of the course itself. Each system is explained in more detail with reference to the relevant mechanisms of information distribution employed.

5.1 Information delivery

The selection of information delivery methods is dependent on a number of factors. These include:-

- The material itself
Each topic lends itself more easily and clearly to certain methods of delivery, i.e. demonstrations as video or computer animation rather than purely still images
- The preparation time and resources available
Each method of presenting the material has its own limitations in terms of format, size, preparation time required, and whether a developer is required to assist in the process
- The availability of the student to view the information
This relates both to location of the student and the facilities available to them for use
- The method under which it is to reach the student
This is dependent on both of the two above items along with student location and having the available technological infrastructure to transmit the material in the desired form.

It is normal for a compromise to be reached, where the tutor is able to provide material in a limited number of formats at a time. This is often due to time constraints, availability of the associated technologies, and whether a developer is available or the tutors are doing the development themselves.

5.1.1 Instructions

Purely instructive information may seem to be the easiest form to handle. The traditional approach to information delivery is to print the information on paper and distribute it to the students. This method, often in text only format, is adequate for some, but not all of the applications to which it has been put. Printed instructions when written well can lead the reader through a step-by-step procedure. The inclusion of diagrams depicting specific actions or the expected output display aids in the understanding of the text and reassurance that the instructions are being followed correctly.

The first step is to determine what form is best for the information concerned, to the current student population, within the present restrictions of available technology and time. Step by step instructions, i.e. a list of actions to be followed sequentially, is one method that copes well in printed form. Others may require the use of diagrams and images to convey their message completely. If so questions need to be asked, and answered, as to “Whether the diagram is best presented in a static or animated form?”, “How much detail is required to explain the intricacies of the apparatus?”, etc. A cluttered diagram containing too much detail can cause more confusion than if it was not included.

When information is presented in electronic form many students are expecting, and often demanding, a greater level of interaction with the material than is possible with printed text. The ability to hold your finger in the book and mark a place is essential along with the availability of an easily accessible glossary. These items are often present in texts that students use, and therefore their electronic equivalent must have at least the same features. Experience has shown that reinventing the book in electronic form is no longer acceptable by most institutions, and definitely not by students.

The way information is laid out will change when implemented in a different medium. Straight text whether in electronic form or not is just that, straight text. When text is presented in electronic form it must be in smaller sections than its printed equivalent. Text should not completely fill the screen, and be should linked to other screens covering additional explanation and related material. This flexibility in navigating and referencing material, be it text or in diagrammatic form can be more easily achieved by utilising the features of an authoring package. These can range from the simple Windows Help or hypertext format to a uniquely crafted interactive package created specifically for that application. There are benefits in each of these approaches, and careful consideration needs to be made before any decision is undertaken to select any particular format. An authoring package should be chosen that matches best the requirements, i.e.:

- Ease of use for the developer
- Produces a maintainable application
- Can handle the various formats the material is to be presented in
- Application can be executed on available computer systems.

Two types of information delivery systems have been implemented and evaluated by their potential users. The first to be described here is a conference information system designed for the use of conference delegates, and the second a tutorial for students in the use of Minitab^{®2}, a statistical analysis package, within the context of one of their courses. The Minitab tutorial will be described in more detail in the next section, Interactive Learning.

Conference Information System

The presentation of information in a static form is analogous to a reference text in which the information does not change. The conference information system presented here is an example of such a static information delivery system. The system was designed to be used by delegates during the conference for information regarding the conference timetable, other activities and events, and the passing of messages between delegates. It was constructed using the Authorware authoring package.

The colours and layout of the information to be presented followed the same format as that of any printed material the conference delegates were to receive; thus saving any confusion between the two systems. The computer system was to be available at several kiosk locations about the conference complex, and to be accessible at any time. This convenience and ease of access to the information, which was updated regularly when changes occurred, was the prime reason for its implementation in computerised form. General design considerations that were not prescribed by the conference organisers followed the pattern as described in Chapter 3, Programme Development.

The component parts of the information system included:-

- the conference timetable
- a brief description of the papers being presented and their authors
- additional activities available to delegates
- a notice board for announcements
- a message passing system between the delegates

Figures 5.1 to 5.5 contain screen prints of various aspects of the conference information system.

² Minitab is a registered trademark of Minitab Inc.

 Presentation Window
X

New Zealand Organisation for Quality
1994 Conference
Developing the Quality Environment

Day 1

Programme		
Chairperson	Ted Martin	
8:00	Registration	
9:15	Welcome and Introduction	Ted Martin
9:20	Official Opening	Hon. Roger Maxwell
9:40	Setting the scene - Developing the Quality Environment	Diane Baguley
10:15	Morning Tea	
10:40	making the Quality Change Happen	David Armstrong
11:30	Measurement of Customer Satisfaction	Dr Simon Sheather
12:30	Lunch	
Chairperson	Roger Atkinson	
1:30	Environmental Management Systems	Prof. Michael Pratt
2:15	Quality Systems Certification, its place and impact in world trade	John Hulbert
3:00	Afternoon Tea	
3:25	Quality from the Seed to the final Product	Mr Dean Witters
4:15	Annual General Meeting NZOQ	
5:30	Quality Society of Australia, New Zealand Chapter Meeting	
7:00	Buffet and Casino evening	

Location: Derek Cole Theatre
Buffet in the Banquet rooms

◀ Page 1 of 6 ▶
Return

Figure 5.1 Conference information system. Full day's programme for day 1; showing what is on, the location of each event, and who the presenter is. The chairperson or each session is also included to aid a presenter in determining who to see before their session.

Presentation Window [X]

File

New Zealand Organisation for Quality 1994 Conference Developing the Quality Environment

Programme	Day 2 Stream 2
Chairperson Lesley Howard	
9:00 The S Mark Product Certification	John Stubbe
9:30 Re-engineering Starts with Competitive Strategy	Peter Salmon
10:00 Who said the Asian Dragons Would Not Buy ISO	Tony Stephenson
10:25 Morning Tea	
Chairperson Gerry Ruffell	
11:00 Critical Component - Managers as Leaders of Quality	Christine Tuffnell
11:30 Quality Function Deployment, a powerful tool or Planning and Development	John Parr
	Phillip Gandar
	Alfred Pang
12:00 Information Systems in a Quality Managed Environment	Gordon Smith
12:30 Lunch	
1:30 Software Quality Workshop	Adrian Price
3:00 Afternoon Tea	
3:25 Software Quality Workshop continues	
5:00 End of Day Two	

Location : Banquet room 2

◀ Page 3 of 6 ▶

Return

Figure 5.2 Conference information system. Programme for one of the three streams that operated on this day. The basic layout is consistent throughout all programme pages.

Figure 5.3 Conference information system. Programme for day 3.

Presentation Window [X]

File

New Zealand Organisation for Quality 1994 Conference Developing the Quality Environment

Programme		Day 3 Stream 1
Chairperson	Adrian Price	
9:00	Good Management Practice	Peter Beddek
9:30	Developing the Quality Environment at Statistics NZ	Gary Dickinson
10:00	Focus, Flexibility & Fulfillment	John Payne
		Richard Long
10:30	Morning Tea	
Chairperson	Bob Harle	
11:00	Customers Revisited - Who Benefits, Who Pays for Public Services	Bryan Wenmoth
11:30	Beyond Wishful Thinking	Gordon Smith
12:00	Towards Real Quality in Education	Dr Nirwan Idrus
12:30	Lunch	
Chairperson	Dan Gable	
1:30	Product & Service Quality - Why NZ must set the standard in its international market niches	Gavin Gormack
2:15	ISO9000 - Is the Gain Worth the Pain?	Malcom Bell
3:00	Closing Summary and Farewell	Diane Baguley

Location: Derek Cole Theatre

◀ Page 6 of 6 ▶

Return

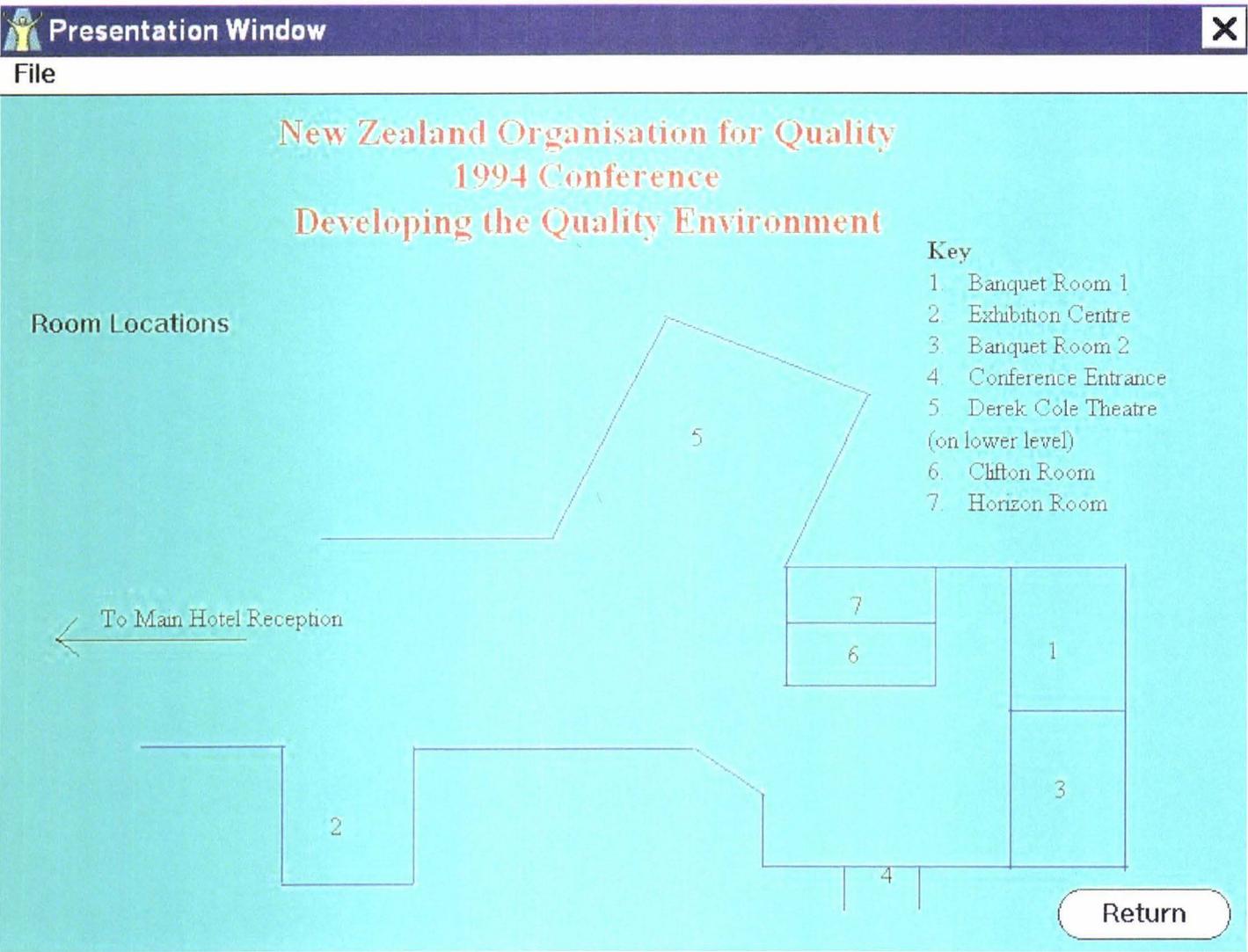


Figure 5.4 Conference information system. Map of locations used during the conference

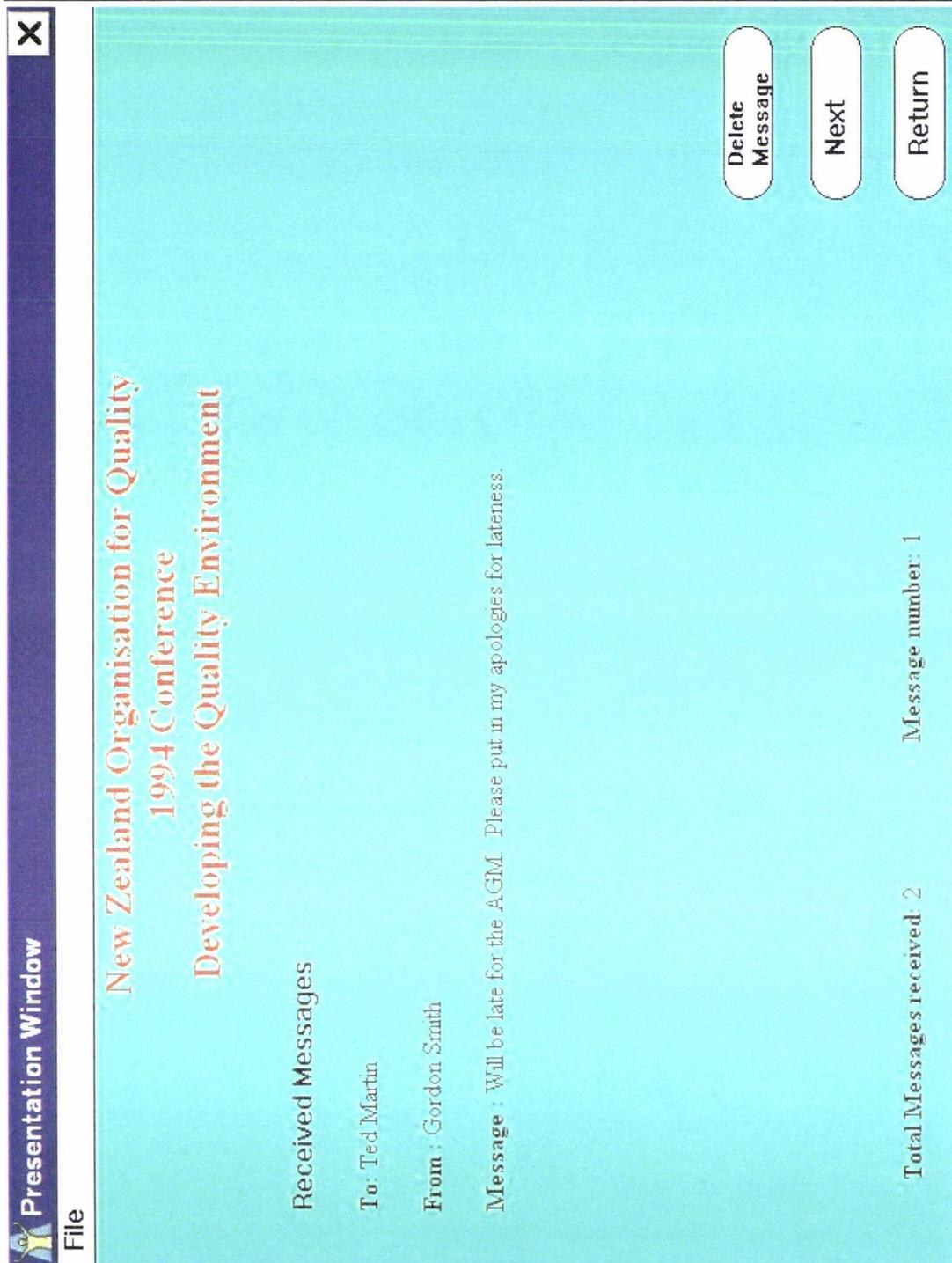


Figure 5.5 Conference information system. Screen seen by delegates when viewing received messages.

The content of this system was relatively easy to handle as it consisted of mostly plain text information which did not change. The timetable was viewable at any time for any day of the conference and defaulted to the current day's programme for initial display. This was done as it was considered most people would be interested in the current day's activities, or the next which was available with a single button push. Paper descriptions were linked to their presentation slots and a delegate could select a description to view from its timetable entry or a list of papers, indexed by both paper and author.

The notice board area was provided for general information, 'conference housekeeping', and announcements to small groups of delegates. These were also made to the conference as a whole where necessary, but as there were several streams of presentations in operation at one time the electronic board was often used by delegates to catch any messages they may have missed. The message passing system enabled delegates to leave each other messages in 'electronic pigeonholes' which were not accessible by other delegates, only the person to whom the message was addressed. Any incoming messages to the conference administration center for delegates were also delivered using this mechanism.

The future use of this system rests in the hands of the conference organisers. It will require someone to reprogram several sections every year and to update the information content. Whether or not the system continues to be used is the responsibility of both conference organisers and delegates. For if the delegates request such a system and continue to use it, the organisers are more likely to persist in its preparation and use. The same principles can also be used for delivery of information about a course of study. Of particular interest is the course outline showing timetable for lectures, tutorials and laboratory classes; the assessment due dates and the weighting each is to have in terms of the final grade. This course outline also contains a brief description of the course and the sections in which it is to be taught. It is possible to include more information in a computerised form than in a printed text form as information is available 'on request' rather than always present as it is in a printed document.

These types of systems have their place in the distribution of information. Due to the nature of static systems it may be viewed as of less use to the reader, yet utilised properly can make a valuable contribution to information delivery.

5.1.2 Interactive learning

As the name suggests interactive learning encourages the learner to be an integral part of the learning experience. This type of learning can be applied in a variety of ways, with or without the use of a computer. Computer Assisted Learning, or CAL as it is known, involves the use of a computer to convey information and solicit a response from the student.

The simplest form of interaction is the quiz type question that has a predefined limited number of answers that can be given. Many students are familiar with this as a multiple choice question. These can form a good assessment of the student's knowledge if constructed correctly, but in most implementations fail to be much more than educated guess work. Their ease of implementation is what educators and programmers have predominantly used as an argument for their continued use. With the development of computer assisted instruction and the desire to utilise a computer in the assessment phase as well as in the instructional phase this has caused a shift in opinion regarding the validity of any method of assessment, especially as it concerns the use of a computer [1].

Short text paragraphs are another form which has been used with pen and paper interactions. These are a little more difficult to reproduce in an electronic environment but not impossible. The difficulty is to duplicate the ability of a human to adjust the assessment of such paragraphs and to judge understanding from the semantics of the text. A computer is limited to key phrase identifications and some

construction. The more the computer is required to interpret what has been written the more complex its programming becomes.

There are more interactions that can occur in an electronic environment that are not often, if at all, used in a standard classroom. This includes the simulation of dangerous or expensive experiments and machinery, which can be simulated by a computer and operated by the simple point and click of a mouse. The interaction used is dependent upon the nature of the subject to be learned and the level to which it is to be acquired. The use of the “point and click” operation with small children is useful and can increase their hand eye co-ordination if implemented correctly. It is also fun for them to learn and thus increases the uptake of the material presented. At higher levels of education, for example at university level, the interaction can take a more elaborate form. The simulation of instrumentation, dynamic systems and experiments being the most often implemented at this level. Some simulations are simple screen representations of the ‘real’ counterpart or can be purely virtual instruments. With an application called Labview^{®3} it is possible for the student to construct a virtual measurement instrument, and then to connect it via an interface board in the computer to an external electrical system. The student’s virtual instrument can then be operated as if it had been a real instrument. Other simulations seek to place the student in an artificial environment that is as close as possible to the real thing, an example of this is the use of virtual reality in training aeroplane pilots, commonly known as a flight simulator. This artificial environment duplicates the controls and interface to the actual plane so as to not form any confusion in the operator’s (pilots) mind. The simulators used with commercial pilots have a physical environment that complements the visual and instrument simulation by providing such things a tilt and pitch of the aircraft. There are many forms of flight simulator from the simple computer based ‘game’ variety to the fully interactive environment used to train commercial pilots.

A simpler implementation of interactive learning is a tutorial exercise where the student is guided from instructions, and helped where required, to perform certain tasks. This can be used where the learning of a procedure for carrying out some function is required, or to teach the student about a concept that is best demonstrated to understand its applications and interactions.

Over the years it has become obvious that many students have difficulty in understanding and applying statistical techniques and analysis to real world problems. One such paper taught within the Bachelor of Technology degree is Technological Mathematics 2, which is centred around the industrial application of statistical analyses to enable the industry concerned to function efficiently and to characterise and optimise its processes. The major tool used within this paper is a computer statistical analysis package called Minitab. At present two weeks of the course is spent on instructing the students on the use of Minitab alone, without any reference to specific applications of its features. This is designed primarily as a mechanism for the students to become familiar with the package while a tutor is available to answer questions. It is desirable for a tutorial on Minitab to be available such that students can work on it in their own time or at home without requiring the presence of a tutor. One such attempt is described below, yet it differs from being a straight forward package operation tutorial (execution of functions within the software), in that it incorporates some of the analysis tasks that the students are required to perform and understand as part of the course.

³ Labview is a registered trademark of National Instruments

Minitab Tutorial System

This system was constructed within the context of a final year student project. As stated earlier the tutorial was designed for use by third year Bachelor of Technology students taking Technological Mathematics 2. It focuses its attention on using Minitab in various areas of Regression Analysis, an area which forms a major part of the course.

The tutorial itself was implemented using an authoring package called Authorware Professional. This authoring tool was chosen for its cross platform implementation, i.e. it is available on both Macintosh and PC platforms. In general, there was little difference except for the screen size, colour and constructs available on the different platforms. This was solved by the specification of two standard interfaces, that were as similar as possible, but used the best available features on each platform. The major design and implementation was carried out on the PC platform as it had the tighter specification restraint in terms of basic system capabilities. For example on the PC platform Authorware takes on the palette of the computer itself and does not impose its own selected palette as it does on the Macintosh platform; so the default Windows 256 colour palette was chosen, this being available on all the PC's that the department currently owned for student use. Both colour and layout of the screen were carefully designed to give a pleasing and non cluttered appearance to the tutorial. Consistency of screen layout is kept paramount as changing it would distract the student from the learning task to trying to navigate around the tutorial.

The structure of the tutorial incorporates as much interaction with the student as is possible without creating too many choices. This is a fine balance, and it was decided to use simple navigational aids throughout the tutorial. This meant that an electronic 'page turner' metaphor was used within the sections of the tutorial with click selection for each section from a main menu display. Figure 5.6 and figure 5.7 show screen prints of the types of help present within the tutorial and the opening 'page' of a particular section of the tutorial. [3]

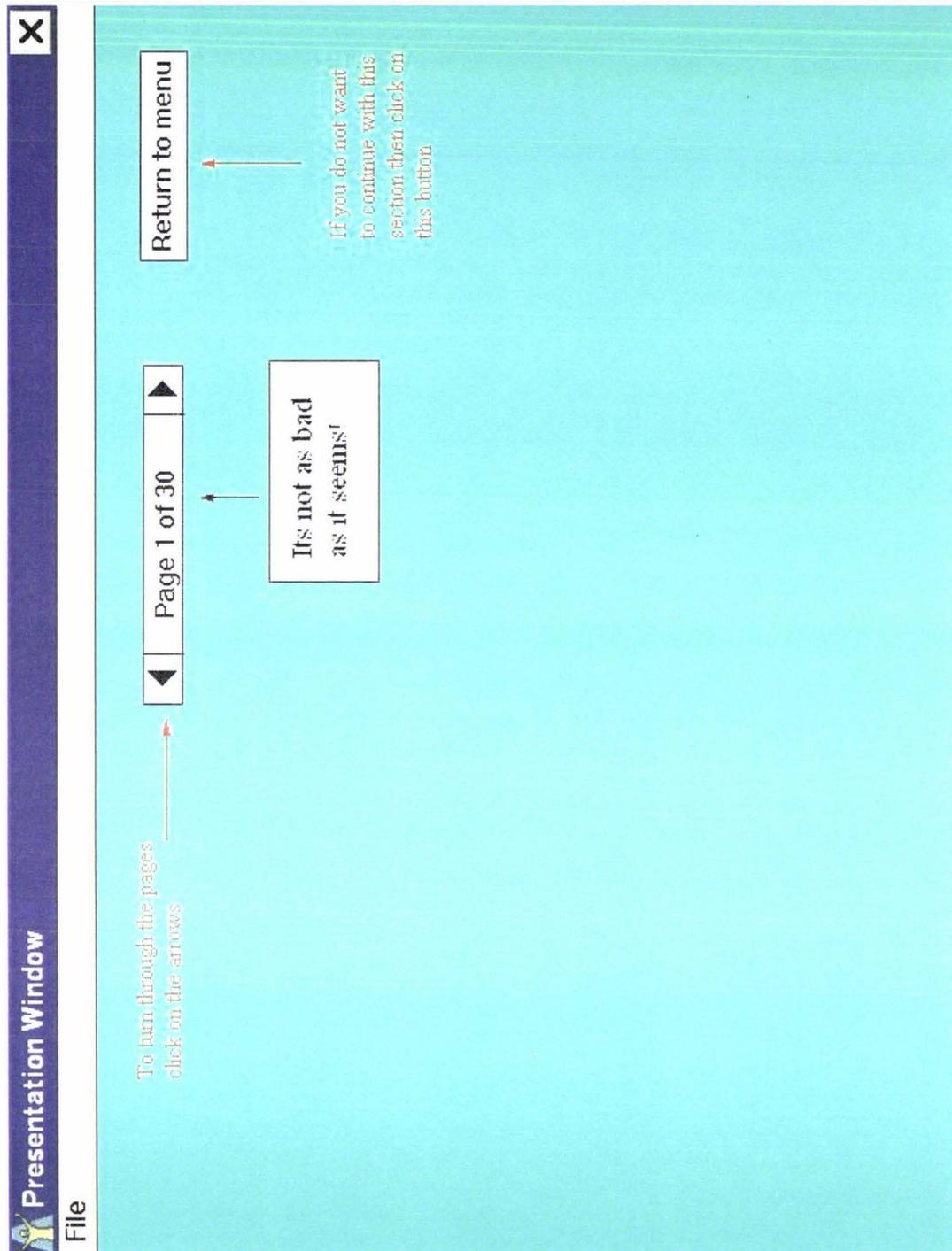


Figure 5.6 Minitab tutorial system. Screen showing assistance given to the student in regard to navigating within the tutorial.

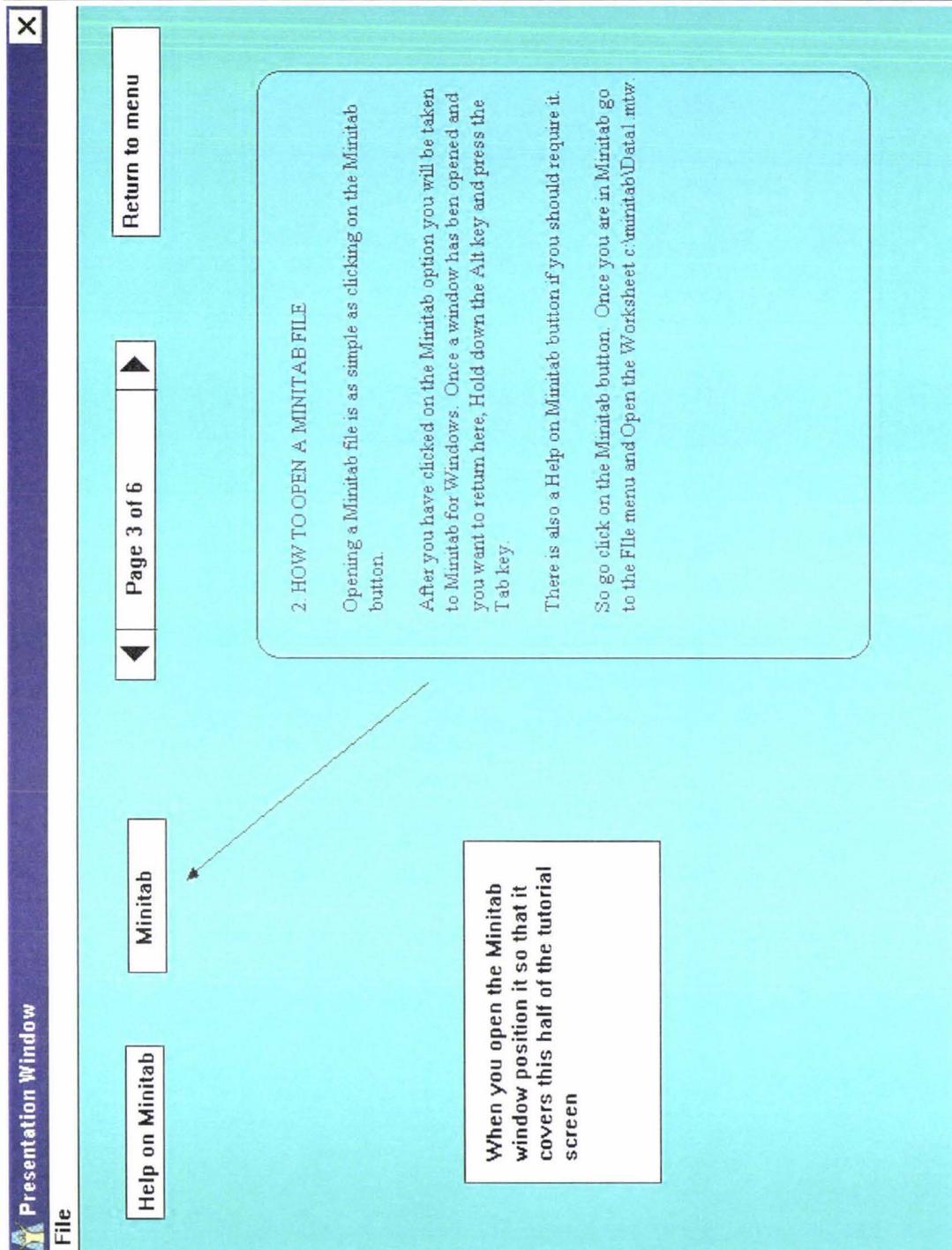


Figure 5.7 Minitab tutorial system. Screen showing the first ‘page’ in the section concerning opening Minitab files.

As indicated in the diagram, students were presented with an introduction to the tutorial which contained navigational instructions (Fig 4.6) along with an introduction to the tutorial material itself. From there students were presented with a menu selection from which to select any of the sections relating to regression analysis. Help is available to the student at any point in the tutorial, as indicated by the ‘Help on Minitab’ button in Figure 5.7. Students are able to return to the main menu and from there to exit the tutorial at any time without necessarily finishing any section.

The tutorial was tested on a selection of the Technological Mathematics 2 class along with some students that had taken the course the previous year. It required the presence of both the tutorial and Minitab to be run successfully. At the point of first testing the tutorial had yet to be made into a 'stand alone' unit and as such required the Authorware package to also be in operation. This setup did not allow the measurement of how well the tutorial would function in conjunction with Minitab in terms of machine time and performance. It did allow the measurement of the response of each student and gauging the level the development had reached in terms of the target users of the system.

Refinements were made to the initial system after this first round of testing, mostly in terms of navigational instructions and trying to improve the context relevance of the help facility when it was called from different points in the system. Not all modifications had been successfully made at the completion of the student's involvement in the project. As the course into which the tutorial was to be placed has changed significantly since the tutorial was first developed, it is to be rewritten to incorporate the new course structure, and expanded to include a self-testing section for students to chart their own progress through their studies within the paper. As development continues this tutorial will become one of many units that will cover the entire course of study within the Technological Mathematics 2 paper.

5.1.3 Integration of tutorial aspects into course material

After this first tutorial was developed the course in which it was used has undergone some dramatic changes. The version of Minitab itself is now different and as a result more flexibility in tutorial maintenance was required. 'How to use Minitab' is still taught as a separate unit, although less emphasis is placed on perfect execution at this point in preference to student's becoming conversant with the way Minitab organises menus and functions and the forms of user input required. Details of specific operations has been integrated into the course material. This gives the tutor the opportunity to explain each function and its execution in relation to its usage and implications for further analysis.

The electronic form of the material is divided into sections much akin to its printed counterpart. When discussing the execution of Minitab commands and their effects, in both the electronic and printed forms, the screen is reproduced (using capture facilities) exactly as it appears within Minitab. A utility application was used to take an image, much like a photograph, of what is currently being displayed on the screen. This image can later be manipulated to eliminate any unwanted clutter leaving only the section of interest. This has improved the understanding and transference from text to practical execution of these operations.

The dedicated authoring package, Authorware, was again used in the production of the electronic version of the course material. As Authorware also has undergone version changes it was possible to use some of the newer features thereby increasing the consistency of presentation and the ease of integration of navigational structures, thus forming a template for presentation and construction of tutorial material. Each section of material is referenced within a "frame". These frames are an Authorware construct that allows the ease of movement between screens, the ability to skip to the start or end of any section, to locate screens using a search function and to jump to the next section as standard features. The availability of these features as standard components allows the author to concentrate on the construction of the material and

not on the navigation structures underneath. The search function is used so that students are able to “look up” definitions and/or to jump to other associated information from their current location. The use of frames also allows the designer, and maintainer, to store the various sections in well defined units and eases the conventional updating tasks that are required with all course materials.

Figures 5.7 to 5.16 depict aspects of this implementation of the course material.

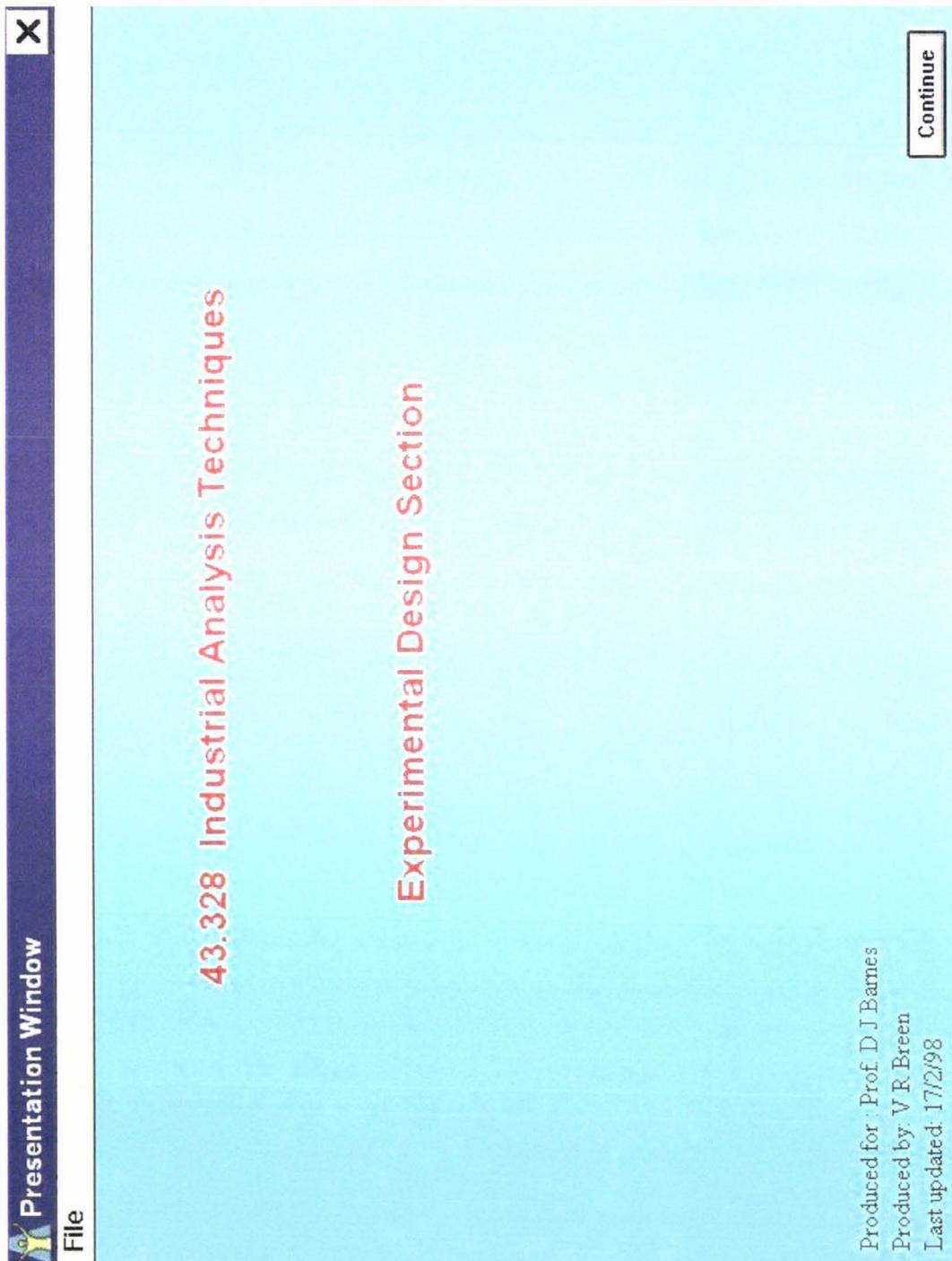
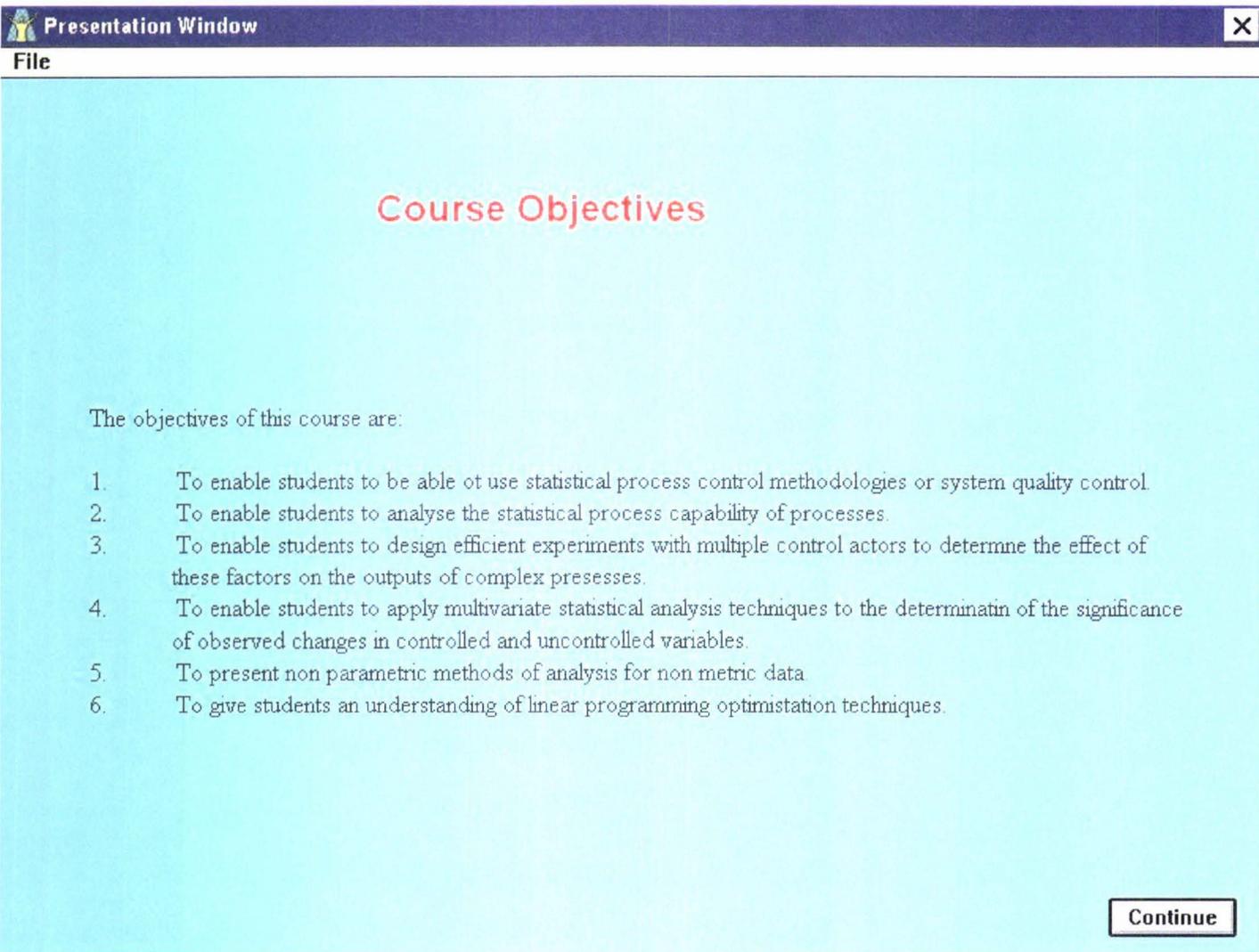


Figure 5.8 Course material. Opening screen.



The screenshot shows a software window titled "Presentation Window" with a standard Windows-style title bar (minimize, maximize, close buttons). Below the title bar is a menu bar with "File". The main content area has a light blue background and features the title "Course Objectives" in red text. Below the title, it states "The objectives of this course are:" followed by a numbered list of six objectives. A "Continue" button is located in the bottom right corner of the window.

Course Objectives

The objectives of this course are:

1. To enable students to be able to use statistical process control methodologies or system quality control.
2. To enable students to analyse the statistical process capability of processes.
3. To enable students to design efficient experiments with multiple control factors to determine the effect of these factors on the outputs of complex processes.
4. To enable students to apply multivariate statistical analysis techniques to the determination of the significance of observed changes in controlled and uncontrolled variables.
5. To present non parametric methods of analysis for non metric data.
6. To give students an understanding of linear programming optimisation techniques.

Continue

Figure 5.9 Course material. Course objectives.

Presentation Window [X]

File

Navigating through this tutorial programme

This box will appear on the bottom right corner of the screen within each study unit. The control functions are explained below. To end this programme select Quit from the file menu in the top left had corner of the menu bar. This is available on every screen.

<p> Return to topic selection screen</p> <p> Go to a recently seen page</p> <p> Find a topic be keyword search</p> <p> Go to the next unit</p>	<p> Go to the first page in this unit</p> <p> Go to the previous page</p> <p> Go to the next page</p> <p> Go to the last page in this unit</p>
--	--

Continue

Figure 5.10 Course material. Navigation instructions.

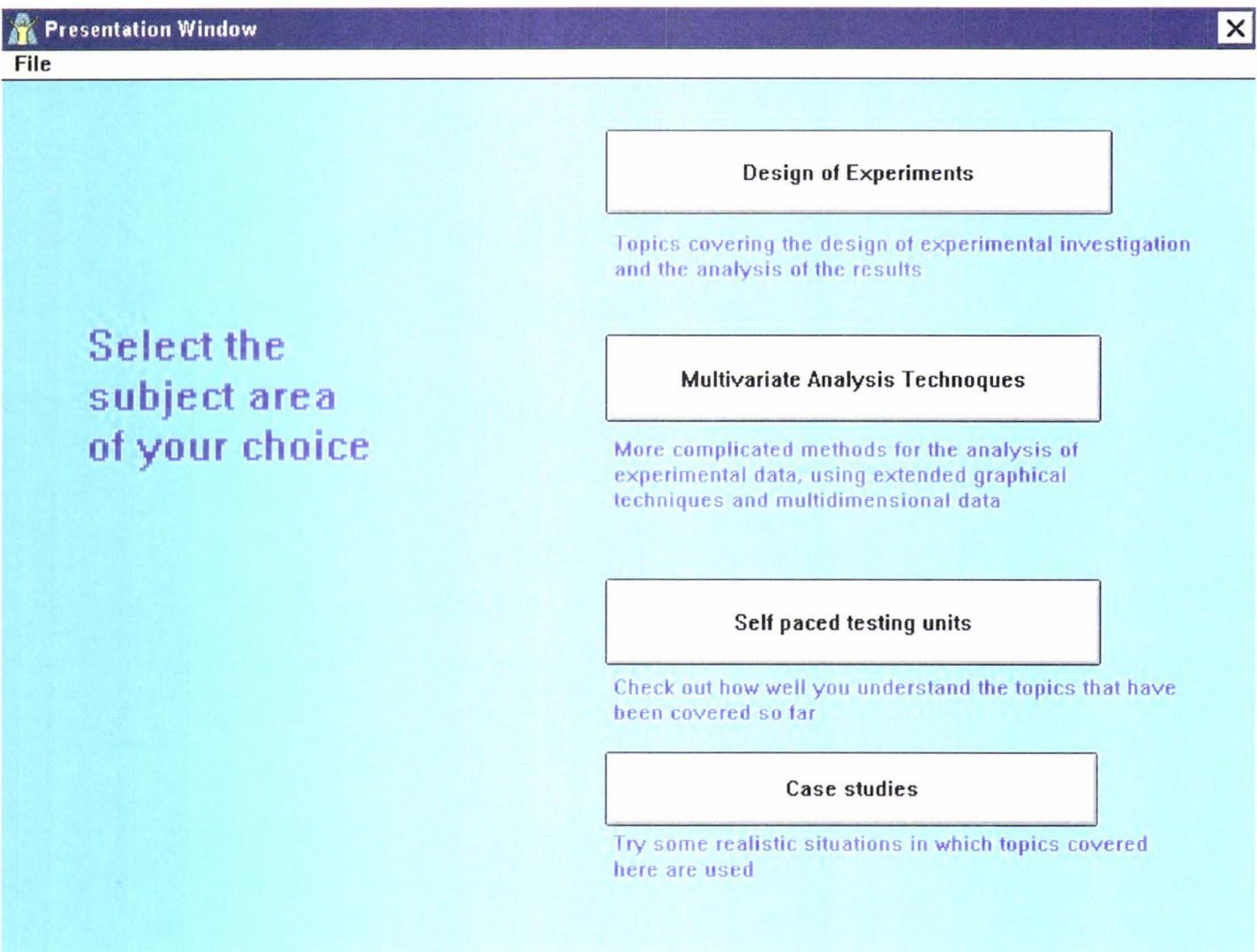


Figure 5.11 Course material. Students are able to choose their area of interest each time the programme is run. No record is taken of the student's progress and so no prerequisites are in place.

Presentation Window [X]

File

Unit Two Full Factorial Designs

This section covers the use of an experimental design, why it is better to use a specific design method, how it is constructed and some simple methods of analysing the results obtained.

There are a number of examples presented here, more are in the printed course material, with some opportunity to attempt a simple example yourself.

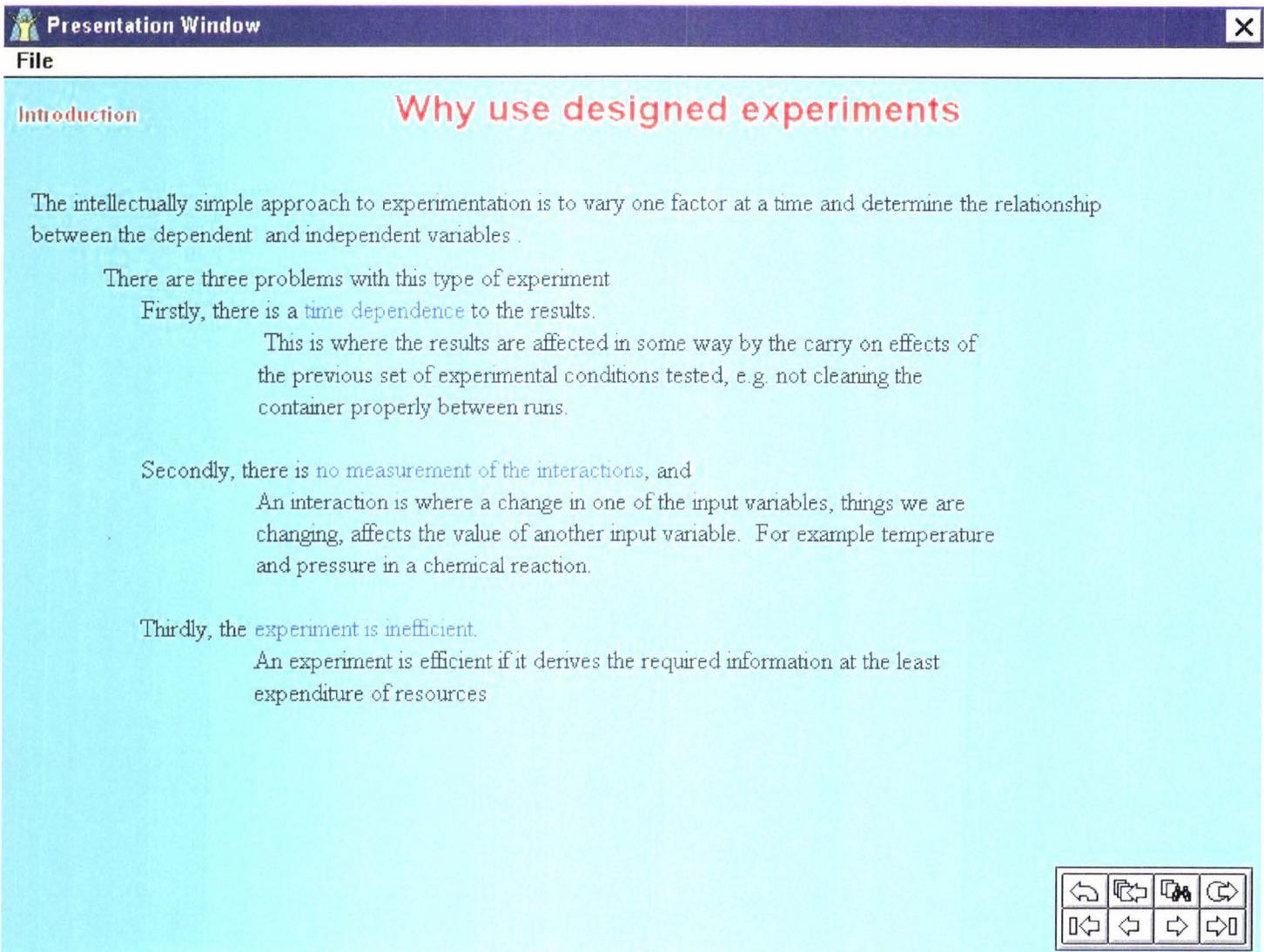
At the end of this unit you should be able to answer the questions found in the self testing section under unit two. Have a go at these at any time, and as many times as you like.

This unit is divided into three sections

1. An introduction
2. The construction of an experimental design
3. Some simple analysis techniques

[Navigation icons: back, forward, search, refresh, home, end, first, last]

Figure 5.12 Course material. Opening screen of the second unit of the experimental design section outlining what is contained in this unit.



File

Why use designed experiments

Introduction

The intellectually simple approach to experimentation is to vary one factor at a time and determine the relationship between the dependent and independent variables .

There are three problems with this type of experiment

- Firstly, there is a **time dependence** to the results.
This is where the results are affected in some way by the carry on effects of the previous set of experimental conditions tested, e.g. not cleaning the container properly between runs.
- Secondly, there is **no measurement of the interactions**, and
An interaction is where a change in one of the input variables, things we are changing, affects the value of another input variable. For example temperature and pressure in a chemical reaction.
- Thirdly, the **experiment is inefficient**.
An experiment is efficient if it derives the required information at the least expenditure of resources

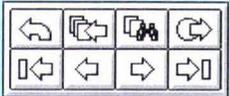


Figure 5.13 Course material. Unit 2 topic on why experimental design should be used as compared to other methods of gathering data.

File
Presentation Window
✕

Construct your own design matrix

Construction

In the spaces provided below construct the design matrix required for a 2 level 3 factor design, adding the Yates code (treatment code) in the left column

Enter the column elements for factor A

Code	A	B	C
▶			

Figure 5.14 Course material. Unit 2 topic allowing the student to enter the design matrix for a 3 factor 2 level factorial design. The screens previous to this one contains an outline of what a factorial design is and how to construct one.

Presentation Window [X]

File

Construction **Construct your own design matrix**

In the spaces provided below construct the design matrix required for a 2 level 3 factor design, adding the Yates code (treatment code) in the left column

Code	A	B	C
	-	-	
	+	+	
	-		
	+		
	-		
	+		
	-		
	+		

Enter the column elements for factor B

Sorry wrong one, try the other level first

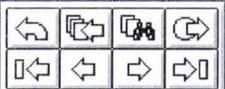


Figure 5.15 Course material. Unit 2 screen showing the response the student receives when an incorrect value is entered.

Presentation Window

File

Construct your own design matrix

Construction

In the spaces provided below construct the design matrix required for a 2 level 3 factor design, adding the Yates code (treatment code) in the left column.

Code	A	B	C
0	-	-	-
bc	+	-	-
	-	+	-
	+	+	-
	-	-	+
	+	-	+
	-	+	+
	+	+	+

Enter the column elements for the code column

Enter the Yates code for this row of values.

Wrong code entered.

Refer to you course text or see you course material for further explanation

Figure 5.16 Course material. Unit 2 screen showing the response the student receives after entering code data incorrectly.

✕
Presentation Window

Construct your own design matrix

Construction

In the spaces provided below construct the design matrix required for a 2 level 3 factor design, adding the Yates code (treatment code) in the left column

Code	A	B	C
0	-	-	-
a	+	-	-
b	-	+	-
ab	+	+	-
c	-	-	+
ac	+	-	+
bc	-	+	+
abc	+	+	+

Congratulations

You have just successfully completed a design matrix with Yates codes for 2 level 3 factor design

Figure 5.17 Course material. Unit 2 screen showing the finished matrix.

5.1.4 Additional help

In many cases students will still require additional assistance from tutors. The computer mediated form of learning is not designed to replace the tutor in any way. The student is free to make use of any or all of the mechanisms at their disposal for the betterment of their education. The tutorial allows the flexibility for students to progress at their own pace and to use different ways of learning best suited to themselves, and this puts a different load on the tutor. No longer can the tutor be two steps ahead of the students in their understanding of the subject in hand, but needs to have a greater overall grasp of the subject from the beginning. All course work done

in an educational institution, such as a university, must comply with certain regulations regarding length and content of individual courses. Yet within this basic boundary the student and tutor are free to move, as long as the aims of the course are measurable and met. The levels of additional assistance given to students will often reflect the alignment of learning styles and delivery mechanisms used within the course. Of greater importance is the assistance needed in relation to assessment carried out throughout the course. Here the bounds placed upon the design and execution of assessment by the educational institution are tighter and the course designers may need to be a little more creative to achieve the desired levels of flexibility within these constraints. Of keen interest to the students is the availability of assistance when they need it. The timely provision of assistance, especially if it is done by a 'real person', can form a major restriction on the degree of flexibility available to the students but it is not impossible to achieve to a satisfactory standard.

A greater use of electronic mail services (EMAIL) for communication between the students themselves as well as with the tutor adds another dimension to the interactions of course participants. Tutors are no longer constrained to office hours but are required to make available electronic office hours as well, although some of this may be done in the tutor's own home after normal office hours.

It may become too great a task for one tutor to be responsible for a single class at one time. The increasing student numbers along with the various delivery mechanisms available can create an overburdening task for one person. Several tutors may be needed. These could be assigned different tasks, e.g. one takes the lectures as normal and others assist in the one-on-one interaction that students require in tutorials and extra sessions, answering EMAIL etc; allowing for the best application of each tutor's individual skills.

5.2 Assessment mechanisms

There are as many ways to assess learning as there are of learning. A match needs to be made between the method of learning and the assessment of that learning. In today's electronic age it is no longer satisfactory to expect students to perform all their assessment with the traditional pen and paper tests. Taking carpentry as an example, not many of us would accept a carpenter's qualifications if, to get them all he had to do was draw the cabinet (or whatever structure you wished). To actually make one, to design it if necessary and to follow the plans, to the satisfaction of a skilled tradesman is what most of us would expect.

The same thing applies when assessing other less 'practical' subjects. As mentioned earlier there are several subjects within the Bachelor of Technology degree that are inappropriate to teach and assess purely as a pen and paper exercises. One such paper addresses issues in the application of statistical analysis techniques to industrial problem solving. The common tool that is used in statistical analysis within tertiary education is a computer package, such as Minitab, as stated earlier. In times past it was necessary to be able to do a larger number of approximations by hand as the computational power to solve them any other way in an acceptable time was not available. This is no longer the case with computer packages such as Minitab which will run on a standard PC, in comparison to older systems that required substantially larger machines. It is now possible for managers to have on their desk the equivalent of what was a mainframe computer several years ago. This increased capacity and availability combined is changing, and dramatically in some cases, the way industry today is functioning. Computer based decision making using

statistical analysis of company data is increasingly being used to improve a company's competitiveness. Within statistical analysis courses for technology students the emphasis has changed from mathematical derivation and theory to the interpretation of analyses of complex situations. The pen and paper test in this instance is no more relevant than the drawing test referred to above for the carpenter. The computer program is now the essential tool which the student must be able to use, were the assessment includes the ability to choose the right statistical method, apply it appropriately and interpret correctly the results achieved.

The major methods of assessment within a structured course still remain the assignment, whether that is a theoretical or practical one, and the formal examination. These two methods are addressed in more detail in the sections that follow.

5.2.1 Assignments

The amount covered in an assignment is a balance between being so small that the assessment is trivial, and being too broad such that it becomes too hard involving a large range of concepts. Most courses are split up into sections or topics which forms a good guide as to the extent of any assignment that is to be set. Within each assessment unit a balance is sought between the theoretical and practical application of the subject under consideration. A check needs to be made as to whether implementing the execution of an assignment in a different manner is adequately testing the student's knowledge and understanding of the subject and is not creating something that is unmarkable by the tutor or unmasterable by the student. Group work and student collaboration is encouraged within the Bachelor of Technology degree. When assessing each student's capability it is important to be able to determine what is collaboration and what is collusion and/or plagiarism. Consideration must also be given to the usage of relevant tools, such as Minitab in areas of statistical analysis, within the assessment process and how specific assignment questions can be set that encourage their use.

Permission was given by the course controller to experiment with the assessment of the Technological Mathematics 2 paper. This paper is taken by all third year Bachelor of Technology students. Experimentation was done within this paper for three consecutive years. In each year the Experimental Design and Multivariate Analysis sections were used, along with Statistical Quality Control in the first year only. The paper has undergone several changes over the three years, but the core topics remained constant. In the same three years the capability of the Minitab package has been enhanced considerably and this has meant that the course has had to be amended to keep up to date with the package. At the same time it has allowed considerable extension to the topics which can be included. Because the analysis algorithms are now available for greater in-depth analysis including the testing of assumptions and so forth, the detail within the course has had to be extended as well. The net result has been to shift the course even further away from mathematical derivations to practical application in increasingly complex situations. This, however, does not reduce the need for understanding of the principles behind the analysis methods. As the associated computer packages for use within each topic have improved and enlarged their capabilities so has the way in which the relevant topics are taught.

The general description of the paper is as follows:

“A practical course in industrial applications of mathematics and statistics for management decision support. A selection

of statistical quality control methods used to understand and control products. Economic experimental design to investigate industrial process improvement including a factorial analysis and response surface methodology. Linear programming to optimise production models in the presence of scarce resources.” [4]

From this description it can be seen that the assignments map to the natural split of the paper’s various areas. Computer packages are available that are suitable to be used with all sections of the paper. In each of the three years of the assessment development project, the assessment within the paper consisted of an assignment per section (3 sections), a terms test (mid year examination), and a final examination. In the first year, two assignments were implemented and marked by computerised means, and one in each of the second and third years. The other assignments were predominantly pen and paper based, with support from computer packages, and both examinations (mid year and final) were completed with pen and paper. The statistical analysis package Minitab was used to implement the computer based assignments. Marking was also done within the confines of the package using Minitab’s own programming constructs. This design decision was made very early in the development to enable evaluation of the assignment implementation in as uncluttered form as possible. A paper was presented at the second New Zealand Postgraduate Engineering and Technology Student’s Conference based on the work done during the implementation of the first assignment in the first year of development [3].

5.2.1.1 Statistical Quality Control Assignment - Year One

The first assignment which was produced required the student to analyse data presented in the context of an industrial process capability study. This type of study investigates the running of industrial processes such as machinery, so that the product produced is within the specification and quality limits. The assessment was designed to cover the following areas:

1. Basic statistical description; including normality of data.
2. Process capability descriptions.
3. Quality control charting.

To enable an assessment to be made of each students’ understanding, it was decided that each student would receive a unique set of data. A macro, (or Minitab program) was to be executed by each student to obtain their individual data. This would discourage copying while still allowing students to help each other in their understanding of the material and assignment completion. Each student received a printed sheet containing a description of the process settings for the assignment, the questions to be answered together with instructions on execution of the data macro, locations in the Minitab worksheet to store their answers, and the use of the Minitab macro language where necessary.

Submission of the assignment was in two parts, firstly a disk containing the Minitab worksheet file, named using the students ID number as the filename; secondly 3 pages of printed graphical representations of their data. Upon completion of the marking of the assignment, students received both printed matter and computer file information regarding their assignment. A printed general response to all students was produced. Included with this was the mechanism used by the marker to generate the master answers for each student. Two files were added to each students disk, one

containing the answers for their data, and the other a text file containing a list of the questions components answered correctly. The graphs were also returned, on which was written the assignment mark.

Tutorial assistance was available to the students throughout the execution of the assignment. Assistance given ranged from executing Minitab commands, to executing the generation macro, to specific areas of assignment material. Not all students required help but its availability was appreciated by those who used it.

Data generation

As each student was to be given individual data upon which to work, a repeatable mechanism to generate the required data points within a known specification was necessary. A random selection of 60 data points was made from a known distribution to give adequate natural variation in the data. To ensure that the data generated was both repeatable and individual to each student the random number generator used was seeded with the last 6 digits of the students ID number. As each student ID number is unique, with the first 2 digits of the 8 digit code representing the year in which the student first enrolled at Massey University, it was a reasonable assumption that the probability of two students having identical numbers was acceptably low. Once generated, the data was placed in a set location, along with other needed information to complete the questions asked. The data generation macro (see Appendix C) was used by the students to generate their own data, and later to regenerate this data for use in the marking process.

Worksheet marking

Once students had completed the assignment tasks and saved their answer file with the appropriate name it was submitted on floppy disk for marking. Automatic marking of the worksheet entailed the development of another macro. This macro (see Appendix C) prompted the student for a code indicating the student's identity, and then compared the student's answers with the generated master answers. Unfortunately, unexpected problems arose as soon as the marking task was commenced, caused by many of the answers received having been calculated in different ways from those expected, and/or to different numbers of decimal places. It appeared that some students had used calculators and typed their answers directly into the Minitab worksheet, rather than use the system to generate them. Specifying the required level of precision before the assignment was issued to the students, would have saved time and reduced the final complexity of the marking macro.

To overcome this problem, the marking macro was revised to mark according to a tolerance band within which each answer was to be considered correct. If the student's answer was not within the required band then, for values that were dependent upon previous figures, the calculations were repeated automatically using the student's own answers as a starting point. According to generally accepted good marking practice, this reduced the flow-on effect of an earlier error or an overall lack of precision. Where comments on the data were required to be made these were numerically coded, providing the student with all possible options from which to select the correct response. (The assignment sheet given to all students is also included in Appendix C.) These comment fields were always recalculated based on the relevant figures, thus judging the student's interpretation of their results without reference to erroneous earlier answers. If the student's answer was found to be within the required accuracy, as indicated in Table 5.1, either initially or after recirculation

the appropriate marks were awarded and comment lines added to the accumulating text file. To take rounding and the precision of display into consideration the initial comparison was more lenient than that recalculated using the student's figures. The recalculated tolerances are shown in table 5.1 in parentheses.

Graph marking

Three graphical representations of the data were requested from each student. Any relevant, specific information obtained from the graphs was included in the worksheet and therefore marked there. All that remained was to do a visual check to confirm the graph submitted was correct.

Question, component	Marking Precision (recalculated)
Q1, data mean	0.0001
Q1, data standard deviation	0.0001
Q1, data correlation	0.001 (0.0004)
Q1, data skew	0.001 (0.0004)
Q1, data kurtosis	0.001 (0.0004)
Q1, comment	exact
Q2, Cp	0.001 (0.0004)
Q2, Cpm	0.001 (0.0004)
Q2, Cpk	0.001 (0.0004)
Q2, Cpmk	0.001 (0.0004)
Q2, comment	exact
Q3, X-bar UCL	exact
Q3, X-bar LCL	exact
Q3, R UCL	exact
Q3, R LCL	exact

Table 5.1 Level of accuracy used in marking

Manual checking

Occasionally a student would receive very low marks from the execution of the marking macro. These, along with the disks of students that the marking macro had difficulty in reading, the file data were manually assessed. Only a small number of disks required this time-consuming treatment, once the marking macro had been modified as described above. On most occasions extra marks were able to be given for correct answers that the marking macro did not register on the initial reading. There was a variety of reasons for this, ranging from numerical data in a text formatted cell, to answers either missing or in incorrect places, to the file not actually being present on the disk. Most problems were easily overcome leaving only four assignments still unmarkable, of the 177 assignments submitted from a maximum of 188, after all these procedures. If extra marks were given, an appropriate comment line was appended to the students text file of correctly answered questions and the mark altered accordingly.

5.2.1.2 Experimental Design and Multivariate Analysis Assignment - Year One

This assignment was to assess the student's understanding of the uses of experimental design and multivariate analysis. As this was the last of the three assignments in the year of implementation its execution relied more heavily on the student's ability to interpret and adequately analyse data that represented true real world situations as closely as possible. To this aim the questions were asked in terms of a scenario whereby students were asked to complete the required analysis and to interpret their results in terms of that scenario. Supporting macros and/or worksheets were provided to assist the students (see Appendix D).

Tutorial support was given to the students as in the previous assignment, but more emphasis was placed on the student's understanding of the situation that faced them and the selection and use of the appropriate Minitab features to complete the required task. This caused some students more trouble than they felt should be required to complete an assignment. Many had significant trouble converting a problem into a series of questions from which to define the tasks required. The assignment was designed in such a way as to stretch the student's understanding of both the material being assessed and its application in more of a 'real world' setting. The interactions between topics were not explicitly stated thus requiring the student to link topics together which reinforces the interaction between them. This is in contrast to the use of individual topic questions which reinforces the idea that sections within the course do not relate to each other.

Each question required the student to draw on different aspects of their knowledge of the subject areas. It could be argued that the topic areas concerned in this assignment were too large for a single assessment unit. This may be true, but only if sections are to be assessed in an individual and unrelated manner. The successful execution of the correctly selected method of analysis showed that the student understood both the purposes and limitations of each method and that they were able to perform the actions required for that method.

It is a constraining factor in any course of study that the assessment of course material be done in the way stated in the course outline and within the specified time frame. Under such constraint it proved necessary to relax some of the normal guides for assignment construction and submission. The due date was made flexible, with the proviso that students could not expect their assignments to be marked by the scheduled return date if they were not in by the submission date. Assignments were accepted up to the time of the final examination. Some aspects of the first assignment were retained, e.g. the generation of individual data for each student. Most of the lessons learnt in the execution of the first assignment were applied in this assignment as much as possible, taking into account the differing scope and design of the assignment.

Assignment Execution

Question one of the assignment tested the knowledge of the use of the analysis of variance test and the related assumptions of validity. Students were required to use multiple tests to assess the validity of the variance test assumptions and the interpretation of the test results. The required responses were entered into the correct locations of the worksheet in the specified format and accuracy, and saved onto a

floppy disk for submission using the specified filename. This procedure of one file per question was used throughout the assignment. Where possible the cell locations for question answers were pre-labelled and formatted for the required response type.

Question two covered the areas of regression analysis and model creation. A model being the mathematical representation of the interrelation of various factors and often used as a prediction equation for process yields. A minimal amount of information was given to the students, again within the bounds of a scenario. The scenario used in this question was one taken from actual industrial experience and adapted for use here. The way in which the question was posed represented as closely as possible the interaction that could be expected from an employer or colleague. The vagueness of the question meant that students needed to be on the lookout for all information, not just that which was immediately obvious to them.

Question three required the identification of model error from a graphical representation of the data. Questions two and three were the most closely related within the assignment. The students were required to enter into the worksheet the numerical representation of the correct interpretation. Three responses were required and six possible solutions suggested. The six possible responses were constructed to test the student's ability to interpret the data while not giving additional assistance to the student. The closeness of some listed responses to each other, and the correct response, was designed to reduce the likelihood that students could select the correct response from the proposed responses by pure guesswork alone.

Question four tested the student's ability to use the Minitab functions to generate a experimental design to be used in laboratory testing. Certain constraints in the scenario meant that the production of the required table was not a trivial exercise as some students first thought. When all the required information was extracted from the question in the correct manner this task proved to be relatively straight forward. Once the design was complete a macro was executed which simulated the execution of the laboratory tests. The students were then required to analyse the results giving the appropriate model values obtained. Graphical representations of the data were also required in some questions.

Marking

The marking of each question was treated separately. Question one was marked by the automatic generation, or regeneration, of the student's specific answers using the exact methods required. These were then compared by eye with the student's results as an automated comparison was not possible within the Minitab programming environment. The remainder of the questions were all compared to standard answers as they were exactly the same for each student. Most students were already accustomed to the concept of individualised data generation from their previous experience with the style of data generation used. This had created in them the almost automatic response of "I'll ask questions so that I understand, not just copy my friend's work", which is exactly the situation we were trying to create. The thought of final examinations looming also contributed to the student's sense of needing to understand for themselves and not rely on other's understanding.

No direct response, except for the appropriate mark, was given when the assignments were returned. This was done to allow the students that submitted their assignments on time to have them returned without enabling the remaining students to copy directly from the master answers. A more detailed response can be given only if all

students were to receive their marked assignments at the same set time. Due to the proximity of the final examination and the flexibility permitted with the submission date this was not possible. It was possible for the student to talk over their assignment results with the tutor or marker on an individual basis if the student wished clarification on their mark and how it was assigned. Enough detail was given to assist the student in understanding the assignment and the material it covered without advantaging any student still completing the assignment. This was an exercise in balance of information and was dependent upon the needs of each student.

5.2.1.3 Basic Statistics and Design of Experiments Assignment - Year Two

In year two, the Technological Mathematics 2 Course had been rearranged in relation to when the different sections of the course are presented, because of the availability of the lecturing staff concerned. It was still basically the same in terms of content, allowing for the difference that occurs when different lecturers are responsible for some sections. Where this assignment occurred in the course was after a revision section on basic statistics in preparation for the design of experiments and statistical process control sections of the course, and towards the end of the design of experiments section. The assignment was issued to students part way through the design of experiments section of the course and due shortly after its conclusion.

The assignment consisted of two sections, the first containing two questions on basic statistics, and the second containing three questions on design of experiments. As in previous assignments the questions posed to the students were written in terms of problems to be solved or information to be obtained from analysing sample data. Enough information was available within the questions to enable the student to identify what was being asked, what information they had, and which processes are most likely to be used to answer the question.

Assignment execution

As in previous assignments each student was given individual data on which to work. The same basic mechanism used in the Statistical Quality Control assignment (year one) was adapted to generate student's sample data for the section on basic statistics. (See appendix E for a copy of the assignment questions and the macro files used.) For the Design of Experiment section the generation of data was done differently. After selecting the right design type to be used the student executes a macro that gives them the levels of each factor to be used in the experiment. This is done by selection from a list of ten different sets of levels (numerical values for each factor) dependent upon a calculation performed on the student's ID number. The last six digits are used, of the eight digit ID number, and in this case the last digit is extracted. The student then created a treatment table, a list of numbers representing the values that the various factors are to have when the experiment is carried out. After this another macro was executed which simulated the execution of the experiment. This included an allowance for experimental error which was dependent on the student's ID entered earlier. Analysis was then performed on the experimental results and comments made. At the completion of each question the corresponding worksheet was saved onto a floppy disk for submission.

To complete each question correctly the student must relate the theory learnt during the course work and lectures to the situation posed by each question. Students must consider what they are trying to investigate, how much of the necessary materials are available for experimental use, the type and amount of data required to correctly perform the selected analysis once the experiment has been carried out, and the purpose for each design method when deciding upon which experimental design technique to use. Each design reflects the conditions under which a series of tasks are to be carried out. When the individual tests are grouped together they form the execution of an experimental design and are often referred to as a single experiment execution.

Marking

The marking for this assignment takes the same basic format as the previous two assignments. Each file was opened and a macro executed that checked the validity of the student's results and allocated marks appropriately. Some manual adjustment of data was needed in cases where students have, for whatever reason, presented data in a format that the marking system cannot recognise. The feedback given to the students was in terms of a paper copy of their results for each question and aggregate mark. A graph was included depicting the frequency distribution of the class marks. The reverse side of the page contained the general outlines for each question and suggested execution steps. To prepare individual master answers for each student in a class of 210 students was considered too great a task within the time limit.

The return of the assignment and mark allocation was not received well by all students. A meeting was called between the staff involved and the students. Several areas of possible ambiguity and misunderstanding were identified in the way the questions were presented. Crossed communication between the tutor assisting with the assignment and the lecturer saw a conflict in the marking schedule for one question. Taking this all into account students were able to resubmit their assignments for marking. The marking system was modified to reflect the issues identified and more manual checking was employed where necessary. These issues were addressed in an expedient manner, whilst still working within the bounds of the course. Lessons learnt here have been incorporated into the design process used when constructing and testing assignments developed for computer execution and marking. The students that felt they were at a disadvantage were given additional opportunities to demonstrate their understanding and these were taken into account in the final grading of the paper.

5.2.1.4 Experimental Design and Multivariate Analysis Assignment - Year Three

It was decided in this year of implementation to integrate the design and analysis portions into one scenario. The students were presented with a process which had been taken from an industrial report on a process, of which they had little specific prior knowledge. The process itself was not changed and the ranges for each element or factor in the process was retained. The students were required to integrate the process of determining which factors contributed significantly to the overall yield, with the determination of the optimal settings for those factors.

The process was simulated within the Minitab package with the use of macro files. A series of equations of similar format were used to simulate process execution.

Five separate equations were developed and one assigned to each student based on calculations made on the student's ID number. Experimental error was added to each execution in a random manner simulating a normal variance in process yield. Each simulation contained 15 relevant factors, three of which were quadratic and the remainder linear with three of these 12 resulting in quite small changes in process yield. As would be expected the quadratic factors contributed the most to the total yield, yet a true maximum yield could not be obtained unless all 15 factors were included.

Assignment execution

For the multiple macros used to function within the limitations of the package, at the commencement of each Minitab session the command SWAP was required to be entered by the student. This command was necessary as the memory management employed within Minitab is insufficient to be able to cope with both the number of and size of the macro files. The command was entered once at the beginning of each time the Minitab package was started. It was not possible to write this command in one of the macro files so that the students did not need to enter it. This is a limitation of Minitab which does not allow some commands to be executed from within macro files.

Students were required to select an appropriate experimental design, create the corresponding treatment table, and execute the simulation control macro. The control macro (Expt.mac in Appendix F) requests the student's ID number and using all eight digits selects and executes another macro containing the process equation to be used. A random error factor is then added to each set of experimental conditions and the accumulated results placed in a pre-labelled results column. For correct execution of the simulation the student was required to label an empty column 'Results', and another column 'Factors' which was to contain a list of the factors that are being used in this experiment. The columns in the treatment table were also labelled with the corresponding factor name. This allows any ordering of any combination of factors to be used without the necessity to modify the simulation macros. Once generated experimental results were analysed to determine the effect of each factor and its associated value. A series of experiments are needed to determine which factors are significant and the determine their optimal value. The steps required to be taken by the student and the understanding of the analysis carried out maps directly the procedures used for similar exercises in industry. It also gives a true assessment of the student's understanding of the material, the interpretation of the results and the inter-relation between various aspects of experimental design and analysis.

Once the optimal values of the selected significant factors were determined, the student submitted a Minitab worksheet on a floppy disk containing a list of factors and their corresponding optimal values in separate columns. A printed copy of the experimental designs used was also submitted to demonstrate the process each student had gone through to determine the results reported in the worksheet. As with all previous assignments tutorial assistance was available for both the assignment material and simulation execution.

Marking

The values entered in the worksheet were marked by a macro. The student's ID number is requested to determine the simulation equation used and which in turn dictates the significant factors and their optimal values. A 20% margin for each factor was used to negate any effect introduced by the random or experimental error

generated at the time of simulation execution. Marks were allocated for first determining each significant factor and then for the correct identification of the optimal value for each. Locating and determining optimal values for the non-linear factors was more difficult, therefore this was reflected in the mark associated with these.

A visual check was made of the submitted printed material to assess whether the student has followed a logical sequence of investigation. Evidence was easily identifiable as to the use and order of the various experimental designs available. If incorrect or incomplete decisions are made by the student this can be seen in the choices made in following designs. This was indicated by the type of design used, the factors included, and the values chosen for each factor. The marks were split with 15% of the total coming from the process of investigation and 85% from the correct determination of factors and optimal values.

5.2.2 Formal examinations

Currently the final examination, if one is present, in almost all university papers is in the traditional pen and paper method. The only notable exceptions are that of oral testing within language papers. As stated earlier each form of assessment must be matched to the subject under consideration, and considered appropriate by the industry to which such skills and knowledge will be later applied. Very little provision is currently made to incorporate the needs of industry into the examination methods used. Some consideration is occasionally used within the selection of topics to be taught, but this is up to the individual institution to decide upon.

The methods used in final examinations should, like their internal course counterpart the assignment, be prescribed by the nature of the material to be examined and its appropriate application. Pen and paper may be appropriate for pure regurgitation of knowledge examinations, yet mostly this method is not adequate in the assessment of the understanding of that knowledge. It is no longer acceptable by most employers that students prove they know the information but they should demonstrate that they can apply it, and that they really understand what they have learnt?

The increasing use of computer mediated education as the method of delivery of material and assessment of its understanding throughout the duration of a course of study, leads to the conclusion that we should be using it more in the last remaining aspect of assessment; the final examination. In time to come exams will be sat in computer networked rooms, at varying locations. Invigilators will still be needed in some form to ensure that the right student is sitting the correct exam and to ensure as much as possible that students have a comfortable environment in which to work, and manage the execution of the examination.

5.3 Macro Design and Construction

At the commencement of the development phase that has spanned three years several design decisions were made which governed the approach and implementations adopted. The one decision that came to have the most bearing on the design and construction of the simulation environments was to establish everything within Minitab. This was so that students could take the assignment home to work on if they owned a copy of the Minitab package, thus freeing up some of the limited

resources on campus and enabling a more flexible approach to be taken to assignment execution. The decision to implement everything within Minitab related not only to the assignment questions but to data generation and marking also. The other major guiding design decision was to make the assignment questions reflect as closely as possible the 'real world' situation, in terms of its operational outputs and data inputs.

The macro files were created using a text editor as Minitab stores macro files as plain text. Macros are the way in which Minitab allows the execution of sections of commands that do not need to be entered at the command prompt every time they are required. The programming constructs available within Minitab are limited in comparison to other programming environments. It was found that some Minitab commands are restricted in their execution, in that they were not able to be used within the macro environment. An example of this is the need for students to enter the command SWAP at the command prompt in the Minitab session window as it could not be executed from within a macro and was needed to enable execution of the simulations within the assignment used in the third year of implementation. This function set the Minitab memory management system to only load into current memory the macro files as they were needed. If all files were loaded at once Minitab was unable to execute any of them and reported a specific error. When this occurred Minitab would need to be restarted for correct execution, as the memory management system could not properly process such a large overrun in memory allocation.

Minitab macro files have a prescribed layout and format, much like any other programming environment. As the versions of Minitab have been released the macro functionality and availability has increased. In the first year of implementation Minitab version 8 was used, and in the third version 11 was the current version. This resulted in many of the functions developed in one year not being able to be directly used in another. The logic of the design of any feature of the macro files used is transportable to any other implementation. With the release of version 11 came the advent of two types of macro files. These are termed local and global macros. The difference is that variables may be passed as parameters into and out of a local macro, which also has associated with it a local worksheet which is active only for the duration of the macro execution. Global macros do not have this feature but can address areas of the current worksheet (data storage area within Minitab) at any time where the local macro can only do this in relation to the parameters passed to it. For the implementation of the assignments it was chosen to use global macros as the use of local macros, although an attractive alternative, would result in the entire worksheet being passed as parameters. Effectively this parameter passing would create a duplicate copy of all information currently in the worksheet. The macro files are presented here in Appendices C, D, E and F covering all the assignment implementations. The structure of each type of macro used and their development in terms of style and layout can be seen in these appendices.

As further versions of Minitab were released not only did the macro environment change but so did the execution of commands within the package. Often commands would be enhanced in their scope enabling the execution of complex operations from a simple menu selection. Further commands were added extending the range of analysis and manipulation of data available. This increase in functionality also assisted in the development of more sophisticated simulation environments. It was then possible to set an assignment question to investigate a more complex system as the design and analysis tools were now available.

The process of designing the simulation environments all start with the determination of the process to be investigated and the style of question to be asked. The question to be asked is the first thing to be finalised in terms of process to be investigated, the type of output to be measured, and the type of data input that is expected. From this the simulation can be designed. A further decision is made as to the number of different versions of the simulation are required to enable some form of individuality for each student. The exact simulation is determined by a calculation performed on the student's ID number which is unique for each student. Once the development is completed the full simulation is fully tested. Each component is tested as it is completed and the final interaction between the various individual macros is tested upon completion.

Testing incorporates the successful execution of the macro files to ensure the students can correctly analyse the results presented. Further testing is come to ensure that the responses required of the students is markable in the format stated. This is also done by a macro. Textual responses are converted to a numeric format for comparison as Minitab does not have a text comparison function. Marks are allocated according to the marking schedule decided upon with the tutors concerned. If calculations are to be made on existing figures that have not been supplied with the assignment (i.e. the students have calculated these figures in the course of answering other parts of the question), any errors in previous figures are taken into account to minimise the carry on effects of any errors made.

When the assignments are returned to the students feedback is given to enable them to understand the marking schedule and where they have failed to complete the assignment fully. If flexibility is allowed in terms of submission date the type of feedback given will then by nature be restricted in terms of what can be given out without affecting the other students who have not submitted their assignments to that point. Formal answers are released upon the completion of the allotted time for assignment submission. These limitations must still be in place ensure that no student is disproportionately advantaged or disadvantaged by the flexibility within the course.

5.4 Conclusions

From the tutor's perspective the use of computer mediated assessment in the first assignment was successful. This was the first time either tutors or students had attempted an assessment of this nature. In at least one aspect the assignment was a complete success - that of marking time. From previous experience, the average time per assignment was calculated at a probable marking time as 32-33 hours for the 188 students in the class. The total time taken to mark the 177 submitted assignments was 15½ hours, a reduction of approximately half, and a substantial saving in staff time. It is anticipated that marking time will be reduced further with more experience of this type of assessment. The marking time for the third assignment was also reduced, although the same comparisons were not made.

A change occurred in the approach and time allocations for the preparation of assignment material during the course of the first two assignments. More time was required in the preparation of the questions and the associated materials needed to complete them. Submission methods were refined and altered as necessary. The submission of a disk which contains the students files was retained throughout as a submission mechanism that utilised the university network was not available.

A comparison between the first three assignments in terms of time spent in various areas of execution makes interesting reading. All three took approximately 2 - 3 weeks to prepare for release to the students, and all had a tutorial assistance component. The extent of the tutorial assistance given was specified by the tutor concerned. In marking time the biggest differences can be seen, with 15¼ hours, 25 hours, 17 hours respectively for each assignment. This marking time included in it the recording of the student's marks and the collation of the assignments into alphabetical order for return to the students. There is a significant change in approach to the assignment setting between the first and second years. In year one the assignment consisted of individual questions, where year two saw the introduction of scenario based investigations. This change is mostly responsible for the increase in marking time in the second year. As year two and three used the same scenario based mechanisms a more direct comparison can be made, noting the further reduction in marking time.

As the assignment changed dramatically in the third year of implementation direct comparisons are difficult with those previously carried out. A one scenario 'question' was used as apposed to multiple smaller settings used in previous assignments. Aspects to remain constant were the relative increase in preparation time and tutorial assistance required during assignment execution and the reduction in marking time. The total time allocation still remains relatively constant throughout all three years of development.

It was discovered at the completion of this last assignment that the style of implementation had achieved educational goals not specifically stipulated at the beginning. It was found that a greater measure of the understanding of the material by the students was able to be measured with greater accuracy in relation to the true level of individual understanding. Students were unable to 'regurgitate' known facts and be credited marks for this. The principles learnt must be used in the correct manner to achieve any reasonable mark. This is also reflected in the operation of the simulation in the determination of true optimal settings. If the student did not determine the optimal settings for *all* significant factors that would produce the specified target 'output' on every execution they had not satisfactorily completed the assignment and this was reflected in their mark.

This greater emphasis on understanding was also reflected in the unsolicited responses from the students. They felt that the assignment was stretching them, and identified many areas in which further learning was required. From past results and student feed back this is one thing that had been missing from the course in such an explicit way until now. It was identified that a shift had occurred within the course in terms of educational goals and the measurement of outcomes. It was now possible to measure the ability of the student to apply the principles learnt in a manner closely akin to that in which these same principles are utilised in industry. This can be seen in the swing from a tool operation based assessment to an integrated approach allowing the tutor to determine not just knowledge in isolation but how students are able to incorporate the various sections of the course in to a whole. This in turn has allowed the extension of various aspects of the course that were previously kept small as they could not be executed within the supporting computer packages or their inclusion in a fuller form was prohibited by the interlinking with other sections of the course.

Security issues still remain relating to the ability of some students to reverse engineer the macro files and obtain the equations for the process simulation. Measures are in

place to limit this as much as possible, yet are restricted by the nature of the macro files themselves. Minitab stores its macro files in plain text formats and as such are easily readable by even the simplest word processing application. As it is desired that this style of assignment is to become a more permanent feature of the Technological Mathematics 2 paper (now known as Industrial Analysis Techniques) further development is necessary on the issues of security and the possible use of network submission and return of the assignment itself.

Achievements to date

Many avenues have been used in the development and introduction of a more flexible approach to learning set within the Bachelor of Technology degree taught at Massey University. Bringing these various aspects together gives a view of the total impact of the research carried out. Individual aspects have been discussed in this chapter and in the previous chapter which focuses on the developmental aspects of the introduction of flexibility into existing courses. Table 5.2 briefly summarises these achievements.

Introduction of flexibility into existing courses	<ul style="list-style-type: none"> • Development of guidelines for the introduction of multiple presentation formats. • Development of guidelines for the loosening of time constraints. • Development of guidelines for the introduction of flexibility into the types and styles of assessment used.
Material presentation methods	<ul style="list-style-type: none"> • Development and refinement of static presentation mechanisms. • Development and refinement of interactive tutorial material.
Assessment mechanisms	<ul style="list-style-type: none"> • Development and refinement of computer based assessment. • Development of 'real world' simulations for assessment purposes. • Ability to measure understanding rather than the retention of facts.
Educational goals	<ul style="list-style-type: none"> • Shift from a tool operation focus to the understanding of principles. • Less reliance upon fact knowledge and more on application. • Breaking down of barriers between topics within the course and between courses.

Table 5.2 Summary of research achievements.

5.4 References

1. *Michael Russel, Walt Haney.*
Testing and Writing on Computers: An Experiment Comparing Student Performance on tests conducted via computer and via paper-and-pencil. Education Policy Analysis Archives, Vol 5 No 3, January 15, 1997
2. J F Lang
Development of multimedia modules for a training system to be used for industry training.
Final year research report, Department of Production Technology, Massey University, New Zealand, 1994
3. V R Breen, J D T Tannock, D J Barnes
Assessment for computer mediated education
Proceedings of the second New Zealand postgraduate conference for engineering and technology students, Auckland, New Zealand, 31 August & 1 September 1995
4. Technological Mathematics 2, Course Outline, 1995
Faculty of Technology, Massey University

6 Impact of Experimental Work on the Teaching Programme

The completion of a three year trial period has resulted in many lessons being learnt. The development of a system for the construction and utilisation of computer assisted assignments, and a greater level of acceptance by the students for the changes that have been made. The process by which the changes have been introduced, the way in which assignments have been set and the resultant feedback to the students have all undergone a continuous review and improvement process. Many changes have been incorporated from student feedback on the assignment process. Further development is always needed in an ever changing educational environment, and the implementation of computer assisted assessment in no exception to this. A continuous process of improvement should be continued beyond the bounds of the experimental work covered in this thesis.

6.1 Introduction and effects of changes made

Any change in the way something is done causes a consequential change in whatever it contacts. In this case we are addressing the education of future technologists. As one aspect of change is accepted and accommodated by the students and tutors into their model of the educational process, additional changes will be viewed from this new perspective.

As the process of education itself is a constant process of change; meeting, assessing and accommodating new information and the acceptance of that change can cause very diverse reactions. Reported in this chapter is a general reflection of both student and tutor impressions of the effects of changing the assessment process in one paper taught within the Bachelor of Technology degree. The paper concerned is Technological Mathematics 2, taken by all third year students. This is the paper described in Chapter 4 Programme Implementation, Evaluation and Refinement.

The instigators of change within the paper were initially the tutors. The desire was to assess the students understanding of the material, without the process used in assessing the student's interfering with the execution of the assessment. That is, the assessment methodology should be transparent in relation to what is being assessed. The student should not spend most of their time dealing with the system, rather addressing the problems presented to them in a manner that shows their understanding of the underlying material.

The tutors and the students were both new to the form of assessment that was used. Due to this it was decided to follow an incremental development plan. As such two assignments were undertaken in the first year, a larger single assignment in the second, and a full simulation assignment in the third. The lessons learnt on one assignment implementation were used in the refinement of the next.

The ability to utilise scenario based simulations of actual industrial processes and to incorporate the entire design and analysis sections into one assignment has reshaped the way assessment within the paper is being addressed. The issues of knowledge integration and application are able to be demonstrated in non-trivial ways. It has transferred the execution of the assignment from question answering to problem solving which is the stated focus of the course.

The focus of the assessment has shifted further towards a generic assessment unit rather than purely assignments. This will allow the development of a truly flexible course, where the material and assessment are available to the student at all times. This development would also include the final examination, thus allowing a student to complete the course in less time than is currently allowed, although a maximum time frame is still imposed.

The shift in what can be reliably assessed has caused changes in the objectives and measurable learning outcomes for the course. It is now possible to test 'real' outcomes, as apposed to the 'artificial' questions that a tutor is restricted to when a pen and paper assessment is used. These changes are reflected in the course outline, the objective and learning outcomes from which are listed below [1].

“Objectives.

The objectives of this course are:

1. To enable students to be able to use statistical process control methodologies for system quality control.
2. To enable students to analyse the statistical process capability of processes.
3. To enable the students to design efficient experiments with multiple control factors to determine the effect of these factors in the outputs of complex processes.
4. To enable students to apply multivariate statistical analysis techniques to the determination of the significance of observed changes in controlled and uncontrolled variables.
5. To present non parametric methods of analysis for non metric data.
6. To give students an understanding of linear programming optimisation techniques.

Learning outcomes.

On completion of this paper the student should be able to:

1. Select the appropriate control chart to monitor a process, calculate the control limits and determine whether the process is in or out of control.
2. Apply cumulative sum (CUSUM) control charts to the statistical control of processes.
3. Determine the appropriate sample size for the inspection and acceptance or rejection of products.
4. Select an appropriate multivariate analysis technique to determine the relationships in complex interdependent data related it industrial processes and services.
5. Determine the limitations of any particular analysis technique in terms of the assumptions which are made for each technique is valid for the particular type and range of data being analysed.
6. Decide what are the significant factors which affect processes and their yields.
7. Design a screening experiment to reduce the number of variables which need to be considered in a complex experiment.
8. Design an efficient factorial experiment to measure the effect of controlled factors and their interactions on the yields of complex processes.

9. Characterise processes in terms of non linear and linear multivariate equations which relate the outputs to the variables which affect the processes using multiple regression or equivalent techniques.
10. Where appropriate, optimise the process using multi-level experiments and optimisation techniques such as response surface regression methodology.
11. Carry out a linear programming optimisation for a constrained problem.”

6.2 Effects on the assistance and explanation required

The change in style of assignment resulted in several changes to the way tutorial assistance was approached within the course. It was found very early on that additional assistance was required by approximately 60% of all students at some stage during the execution of each assignment. The assistance was both in terms of the understanding of the material itself and the execution of various parts of the assignment. It is not uncommon to provide help for students during assignments in the understanding of the material. The level of assistance given in this regard was comparable with that of other non computerised assignments within the paper and that given in previous years.

Group work is encouraged by the Department in which this paper is taught. This has led to groups of students seeking assistance together on an issue they have previously discussed amongst themselves and have been unable to satisfy themselves that they fully understand. The group approach although not compulsory in this paper is encouraged for the very reason stated above, that students are able to assist each other in the understanding of the material which in turn reinforces their own understanding. When a group of students approach a tutor for assistance it reduces the number of times the tutor is required to explain the same point and it enables the tutor to utilise segments of knowledge held by group members to assist in the explanation, thus drawing out from the students that which is already understood. An additional effect of the group approach has been to encourage further learning amongst the students in that they are proud of the ability to explain aspects of the course to their classmates without the assistance of a tutor.

The execution of various parts of each assignment have caused difficulty for some students. This has been for the most part in the execution of Minitab commands and in the deciphering of the output of analysis. Detailed instruction on the operations that are necessary for correct execution of macro files and other commands necessary for the successful execution of the assignment rather than the actual solution to the assignment problem were given in the printed material that accompanied the written assignment questions. Also included in these instructions was a description of the method for submitting assignment material for marking. Although this information was given in printed form the question most often asked by students, concerned the order and correct execution of the macro files, and the format of answers to be submitted. Many of these aspects were also covered during lecture or tutorial time so that the correct information could be given to as large a number of students at one time as possible. All efforts have been taken to ensure students have access to the relevant information for the execution and submission of assignments, yet it still remains, to the annoyance of the tutor, that a certain percentage of students will either not listen or misplace any instructions given.

A tutorial is held every year prior to the commencement of the sections of the paper in which Minitab is to be used on its use and execution. A common problem encountered is that some students have difficulty in retaining the knowledge gained during the tutorial phase for later use in assignment work. Steps have been made by the tutors to reduce the time spent on the initial tutorial and increase the amount of Minitab commands discussed in the lecture during the associated sections. This has reduced the amount of questions received on basic operation of Minitab when the assignment is released. The approach that has been taken forces students to adjust their perceptions of assignments, from a question answering exercise to a problem solving one. This does not match some students ethos of study and learning. It also forces them to decompartmentalise their learning, thus solving the problem stated above of the lack of retention of information from one section to another.

6.3 Guidelines for future implementation of change in the assessment process

When constructing an assignment question careful consideration needs to be taken with regard to the amount and level of instructions needed. Enough specific information for the student to answer each assignment question must be contained within the relevant question. If for example the student is required to place responses in particular locations these must be explicitly stated within the assignment. It should be remembered that when students are completing an assignment they do not always remember the instructions given in other sections of the course, even if they are relevant to the assignment questions. This necessitates the need for step by step instruction to be given, particularly where the execution of commands within the programme is concerned (in this case Minitab). However, this must not conflict with the requirement of the assignment that the student must exercise good judgement in selecting the relevant technique, executing and interpreting analysis correctly in proceeding with 'the experiment'. A separate section should be included within the assignment detailing the assignment submission instructions. Where a file is to be submitted, either via the network or on a disk, specific details as to filenames and file types should be stated. Assignment questions should contain all required information in a manner that is not ambiguous.

The assignment questions and any associated programmes (macros), will be required to be fully tested before release to the students. Macro files should be thoroughly tested as to their correct operation in both normal, expected conditions and in unusual conditions. If 'error' responses are to be given to the students from within the macros then these should be self explanatory, and/or a fuller explanation contained in additional support material. If the system developed or the assignment is to be executed on a network it should be tested on that network, for example the execution of some packages will be different in a student computer laboratory from that running on the tutor's computer.

Because of the difficulties experienced in the initial development of this assignment it was necessary to explain to the class that the assignment was an experiment and to involve them directly in the development process. They were asked to comment on any shortcomings identified (and to report these quickly so remedial actions could be taken), parts they found helpful in the identification of areas which they did not fully

understand, and those portions they found relatively easy. This had the resultant effect of obtaining numerous comments that were able to be used in the refinement of the assignment, both for the first implementation and future ones, and diminishing the 'hassle factor' from students having difficulty with the assignment. Even so, throughout the assignment execution it will always be necessary to have a person available to answer student queries. These questions often fall into two groups, those concerning the assignment material and those concerning the execution of macro files and other support programmes. Clarification questions as to format will be asked by some of the students as the due date approaches, even though these are contained clearly in the assignment itself. There may be occasion to correct an error or malfunction in the support macro files during the time the students are working on the assignment. At this time it is critical that the error is corrected rapidly and fully tested. Students should be notified of any changes to the assignment execution, as they may have to repeat some sections to counteract the effects of the error. All notifications to students should be placed in a prominent place, which is also stated in their assignment sheets. For example this communication could take place within a news group on the Technology Electronic Forum (TEF), and placed on notice boards within the Department.

The system used to mark the assignment should also be tested before the assignment is released to the students. When the assignment is submitted it is the responsibility of the marker to complete the marking in the stated time. This allows greater time for in-depth feedback to the students and to address individual concerns without interrupting the successful completion of the course. The amount and style of feedback to be given with the return of the assignments should be agreed upon by the tutor and any other associated staff, and stated clearly in the assignment sheet.

When submission of assignments is via a floppy disk then there is a greater possibility of data corruption and file unreadability. This can result from many factors over which the student is not in control. Time should be allowed within the marking period to locate affected students and to allow them to resubmit their assignments. It is a good idea to constantly remind the students that all files used in the completion of their assignments should be backed up regularly, not just the ones being submitted. If it is suspected that a student has not completed the assignment themselves but copied others work, then they may be requested to complete some aspects of the assignment again on the tutor's computer with the tutor in attendance. To date this has only been necessary on two occasions which resulted from the loss of assignment data rather than from the suspicion of cheating.

Above all else when developing computer based assessment materials, the approach taken must be consistent, both in terms of the amount of information contained in the assignment and the responses given by the supporting programmes.

6.4 Outcomes of the experimental work

As is noted in Chapter 4 the marking time for each of the assignments has been reduced from that associated with the implementation of similar assignments in pen and paper mode. The rearrangement of time allocations has not been the only outcome for the experimental work carried out. A greater understanding has been gained of the assessment process, and how the incorporation of appropriate tools into the assessment process can be achieved.

At the outset the ideal was to utilise the statistical tools taught in the course to assist in the assessment of student's abilities to understand the principles of their operation and to be able to use them correctly. For this the student would need to understand each aspect of the subject in terms of the underlying principles and its interaction with other topics within the course of study. This is a shift from a 'knowing the facts' style of assessment to a 'understanding the principles, their interaction, implementation and application' style of assessment.

Over the course of the three year period the style of implementation and presentation of the assignments has undergone some dramatic changes. No longer are individual questions asked relating to only one specific area, but a total interaction and amalgamation approach has been adopted. By the students' own admission this has resulted in a increased awareness of the inter-relation of topics and 'how the whole thing fits together'. The students feel more involved in their assessment and consider that the assignment work has greater importance than just within this one paper. A number of students have commented that this style of assignment has prepared them better for the applications of the subject matter in later employment. A significant difference has also been seen in the use of Experimental Design especially, in final year project experimentation. All Bachelor of Technology students complete a project in their fourth and final year of the degree. The project is a significant part of this year's work and is used to introduce students to industrial style projects, their operation, purpose, and presentation of findings. More students are approaching their experimental work in these projects with better understanding of experimental design and analysis as a problem solving tool or as a research tool for investigating complex systems. This is borne out in the experimental work undertaken in subsequent years.

In the first year of implementation of the changes in assignment work a survey was given to every student that completed the assignment. The response was so poor (13 from a total of 177) that further formal surveys were not carried out. A formal review is still to be done and should be under the administration of independent assessors. Because of this poor response, to obtain feedback from the students, unsolicited verbal comments were used to guide the reception and perception of the assignments themselves. This form of feedback has become an accepted part of the course and students engage in it freely. The increased involvement of the students in the assessment process led to an increased perception of the importance of their participation in the development of the course itself and the assessment methods being used.

As aspects of each assignment became known to be troublesome, they were addressed in an expedient manner. If a student reported trouble with the execution of the assignment macro files this was investigated and any corrections made the same day. The class was then notified at the earliest opportunity. Some issues that came to light were not the fault of the macro file execution but of the student's use of them. This was also able to be corrected when the student was observed in the laboratory or came to the tutor with the files on which they had been working. Some difficulties arose due to a design decision made early in the three year implementation period. That was that all work was to be done totally within Minitab. This is to allow the desired degree of flexibility within the assignment execution, students only required the presence of Minitab to execute the assignment enabling them to work at home if they owned a copy of the package. The simulations implemented at each stage of development stayed within these bounds

and as a result were not always implemented as the developers would have preferred since Minitab lacks a substantial programming interface and language. The macro files are in plain text resulting in their availability to students to interrogate, therefore obtaining guidance if not the answer to assignment questions without having to actually complete the assignment work itself. The programming limitations of Minitab also introduced an artificiality to the simulation process that may have been avoided if implemented in another manner. As a result of the decision to stay within Minitab the developer became proficient in the development of complete macro driven simulation environments utilising all available constructs to do so. The development of further systems is also likely to remain within Minitab for the reason of flexibility stated earlier. With the further development of the Minitab programming environment more sophisticated simulations can be developed.

The quick response to errors in the system, be it electronic or in the written assignment instructions, led to the identification of a number of important issues. A fuller testing of the assignment macro files was required. This was incorporated from the second assignment in year one on, and testing was also carried out in the student laboratories to ensure correct execution in this environment. The level of tutorial assistance required by some students resulted in clearer instruction being given in the written assignment, along with the explicit detailing of assignment submission information. Tutorial support was emphasised in all implementations and became more structured as the assignments developed. Many students would come in groups to ask questions. This was exploited enabling the tutors to assist a greater number of students at once, and to incorporate the knowledge the other students already had. Within these group sessions the partial understandings of each student could be extracted and utilised to assist the group as a whole answer the query with which they approached the tutor. This reinforced the Departmental attitude to the encouragement of group work. When it became apparent that a significant number of students were having trouble with a specific section of the assignment, time was set aside in the next formal tutorial time to address the issues. A number of areas were explained again by the lecturer at this stage reinforcing what had previously been covered and the lecturer's commitment to assisting the students in their understanding of the material. All lessons learnt in one implementation were carried on to the next with an increasing level of attainment of the overall goal of student understanding of the material rather than the regurgitation of facts.

During the first year of implementation the students were informed that the style of assignment they were to complete was under experimentation. This enabled the tutor and lecturer to involve the students in the process and to elicit a quicker response from the students if anything was not clear, or was not working as it should. In later years, and especially in the third year the students were not informed that the assignment was anything different to that which they would normally be receiving. Thus unaware of any experimentation this group of students formed the first 'blind test' subjects. Their reaction to a computer based assignment was one of acceptance and upon completion, one of understanding what was intended by the tutors in setting this style of assessment. Again unsolicited comment was made to the effect that a greater level of understanding had been reached of the subject material, and its interactions and implementation. One student that was repeating the paper after having failed it in the previous year made comment to the lecturer that he understood better than he had done the year before, even though the material had not changed.

As each implementation of computer assisted assessment was from different areas or covered a different range from the same areas it is difficult to judge the level of

student understanding from a pure comparison of assignment marks. A truer comparison can be made using the results of the two examinations within the paper, the terms test and the final examination. These have showed an increase in the average raw marks gained by the class. During the third year of implementation the paper in which this work has been carried out was taught twice, once on each of the two major campuses of the University. The increase in raw marks was so marked for one group of students in this year that grade point average was effectively raised by a considerable amount from this paper alone.

6.5 Future implementations

The level of flexibility that can now be obtained within a course using the teaching and assessment methodologies outlined in this thesis can be clearly seen. It is now possible to allow students to work where they choose, at the pace they prefer, and to undertake formal assessment at a time chosen by themselves. If a student is capable of passing the course in a shorter time period this benefits both the student and lecturer. The student is able to place greater effort and emphasis on other areas of study and the lecturer is free to assist those of the class not so quickly conversant with the material. The ability to study, and especially to complete assessment work, at a location other than the University itself takes pressure off the already limited facilities allowing other to obtain full benefit of those resources.

The way in which assignments in the areas of Experimental Design and Multivariate Analysis within the Technological Mathematics 2 paper, or Industrial Research Techniques as it is now called, is to continue. This decision has been made by the lecturer for these areas of the paper, and a commitment made to the further development of this style of assessment within the paper. The topics addressed to date in the assignment work only cover part of the entire course. It is possible to expand the implementations to include, probably in further assignments, other more complex situations and analysis techniques, for example the increased use of non-linear techniques and Taguchi analysis. This will form an important part of the total assessment process, especially as it is important for students to understand where each technique is appropriate and how it is used. Currently this is not fully tested in any format, assignment or examination. It can be concluded that the methodology adopted within the assessment areas of the Technological Mathematics 2 paper could be extended to assessment of student competency in these areas as well.

Other subject areas may also be applicable to this style of assessment and investigation in to its use is needed. The trials that have taken place in the Technological Mathematics 2 paper have shown the validity and success of such methods to other staff within the Department. The process of assignment design and implementation is now stable enough to allow its portability as a technique to other subject areas. Implementation in other areas is reliant upon the support of the lecturing staff concerned and a commitment from the Department as a whole to adopt this style of assessment. A caution should be given that not all areas of study can or should be or need to be assessed in this manner. The use of appropriate tools in the assessment of student understanding was the aim of this experimental work and it remains the chief guideline in the implementation of any assessment methodology.

As within the Industrial Research Techniques course itself, continuous improvements are being made to the assessment methodology developed. As stated earlier it is now

at a point where it is stable and can be applied to other areas of study. Further work is planned to implement the style of assessment used here among assignments in the execution of a final examination run under the same regulations as all university formal examination. A generator is also under development to aid in a more robust and semi-automatic generation of the macro files required to support the simulations and marking of assessments set in this way.

6.6 References

1. *43.328 Industrial Analysis Techniques - Course Outline*
Faculty of Technology, Massey University, 1998

Appendix A Courseware development tools - evaluation checklist suggestion

This check list comes from Philip Barker's book *Authoring Languages for CAL* published by Macmillan Education Ltd in 1987. It suffers a little in relation to the developments in hardware and software systems since publication. It still offers a good reference to developers and authors as to the value and durability of the systems which they create.

1. Is the system easy to learn?
2. Is the system easy to use?
3. Does the system allow easy amendment of courseware?
4. Are different levels of authoring allowed?
5. Can the author easily create screens of text?
6. Can multiple text and background colours be used?
7. Can the author define personalised character sets?
8. Can multiple fonts and character sizes be used?
9. Does the system provide facilities for audio support?
10. Does the system provide facilities for graphic support? (Slides, video tape, video disc)
11. Does the system support computational graphics?
12. Can text and graphics be freely inter-mixed?
13. Can text and video be freely inter-mixed?
14. Can computational graphics and video be mixed?
15. Can video from different sources be mixed?
16. Is colour easy to use?
17. Does the system permit tactile control/communication with other instructional aids and devices?
18. Can the software systems be accessed during an instructional session?
19. Is it possible to attach the courseware development and/or delivery workstations to a network as a communications link?
20. Is the system micro-computer based?
21. Is the system available on a range of microcomputer systems?
22. Are mainframe/microcomputer development facilities available if needed?
23. Can courseware be developed on a mainframe and delivered on a micro?
24. Is the system well supported?
25. Is the system well known?
26. Is the system widely available?
27. Are there consultants available should the need arise?
28. Is the documentation easy to follow?
29. Is the documentation well written?
30. Is there an on-line help system for authors?
31. Can development and delivery modes be rapidly switched?
32. Are there good facilities for debugging and testing courseware?
33. Does the lesson strategy allow branching?
34. Can the student recap?
35. Is there an interrupt/restart facility?
36. Does the system offer good computational facilities (for simulation and numerical work)?
37. Does the system provide good answer checking facilities for both numeric and textual responses?

38. Can student responses be filed away for subsequent processing?
39. Does the system provide support for input from a variety of devices? (Keyboard, touch screen, touch pad, light pen, mouse, two mice, tracker ball, joystick)
40. Does the system provide facilities for the control of multiple displays?
41. Can the activity on these screens be coordinated in various ways?
42. Does the system enable the creation of student/trainee models?
43. Can the system produce software for use in a group learning situation?
44. Does the system permit the use of a database?
45. Does the system allow the use of a knowledgebase?

Appendix B Authoring Languages and Environments

ACL	Author Command Language is a special applications orientated authoring faculty that is used in the preparation of instructional material for use in keyboard skills training.
ADAPT	ADAPT is the name given to the authoring faculty that is provided with the MACRO TICCIT CBT system.
AECAL	Audio Enhanced Computer Assisted Learning. This is a technique whereby sound is used to augment a conventional computer based learning activity.
AGRAPH	This is a graphic support system that can be used to generate material for use in a CAL environment. AGRAPH is a data-file orientated easy-to-learn and easy-to-use graphics package that can plot a variety of different types of graph using a wide range of display devices
APL	A Programming Language, this interactive language is not really a CAL\CBT authoring system as such; although it has still been used.
ARTPLAN	Auto Rover Programming Language, is a programming system for controlling interactive conversation with a computer. It is based on a variant of the PILOT system (called GPILOT). ARTPLAN itself however is implemented in the 'C' programming language.
ASET	The Author System for Education and Training is an instructional system that was designed for used on large Unival mainframe computers.
AUDIOTEXT	This system is not so much an authoring facility but more a novel linguistic form. Its development is based on the belief that the textual and source representations of a message are can be quite different. The system has been used primarily for providing audio support for handicapped persons. However it could be used for many other applications where spoken messages are required.
AVCAM	Audio Visual Computer Aided Tutor is a multi-media training system that uses slides and audio recordings as the primary support media within a computer based instruction workstation.

BASIC	Not a CAL language as such although systems have been developed using it as it is so simple to learn.
BOOK	The book system is a Cal authoring and presentation system that has been developed by Otsuki and Tukeuchi at Kijushu University in Japan. It was created in order to three basic requirements. First to allow students to study freely and independently (searching and accessing Cal based materials to facilitate the solution of a learning problem): second to ease the burden of producing CAL materials: and third, to put a large collection of instructional material into a form that could be easily managed and applied automatically.
CAN.8	It is a fairly standard CBT system allowing instruction to take place through the medium of text and graphics (which may involve video tape input or the use of video disk), However it can be extended to advanced technical training applications by using other peripherals such as robots and machine tools.
CATS	Developed essentially as a prototype system, CATS is a highly interactive, graphics - orientated interface facility for use in Computer Aided Videodisc Education System (Cave) It is an easy-to-use authoring system that is intended to be used by teachers within an interactive video-based learning environment.
CAVIS	Computer Audio Visual Instruction System is an interactive video system. In contrast to most IV systems CAVIS is tape based. The essential features of a CAVIS workstation are a video cassette recorder/player, CRT, a specially designed keyboard (for student interaction), a floppy disk system (for program and information storage), and a controlling computer.
CICERO	AN authoring system developed in order to generate tutorial CAL in a distance learning situation (open university). The courseware produced by CICERO consists essentially of frames of textual information interspersed with multiple choice questions. The system can provide students with diagnostic feedback and remedial help. It can also be used as a revision tool.
COMBAT	A CRT based authoring system for computer assisted learning. Can be programmed using existing templates or UCSD-Pascal
COMET	COMET is an interesting example of a microcomputer based menu driven authoring system. It has been designed to cater for the specialised needs of foreign

language instruction. The basic system is designed in a highly modular fashion and as such a way that is easy to produce version of the package for a variety of foreign languages. COMET is an abbreviation for Computerised Modular Exercise Templates. The term template is used to designate a driver program containing a previously established instruction pattern. COMET is orientated towards interactive language practice rather than just information presentation. COMET is implemented in BASIC and uses a variety of sound, light and motion effects to handle student errors. Another useful feature is the ability to produce foreign characters.

COURSEWRITER

This is a language that was developed many years ago by IBM for use with their early computer-assisted instruction equipment. The system (as COURSEWRITER 3) is used as a basis for a number of other instruction packages eg IIS

CYCLOPS

CYCLOPS is an audiovisual distance teaching system that was developed by the Open University during the late 1970's. The first version of the system was based upon the use of a standard TV set and an audio cassette player - the latter being embedded within the CYCLOPS interface box. Both the audio and visual information were stored on the audio tape which was played back using the CYCLOPS unit. The output from this was fed to the aerial socket of the TV set for viewing by the students. Since the release of the earlier CYCLOPS system a number of enhancements have been made. Thus by providing light pens and communication modems at the student stations it is possible to convert the basic passive CYCLOPS system into a highly interactive telewriting system for use in distance tutoring.

DECAL

The DECAL system was developed in the late 1970's by the Digital Equipment Corporation (DEC) for use with their popular PDP/11 range of minicomputer system. The system itself is written in BASIC and can be used to produce CAL in virtually any subject area. There are 15 built in templates that enable a courseware author to structure a lesson according to its intended application.

DIALOG

The DIALOG system is a screen-orientated, menu driven system for handling instructional CRT frames. It is written in Pascal and in many ways is very similar to the COMBAT system. A variety of different types of CRT screen function can be implemented.

- DOMINO** DOMINO has been described as: an 'elections hook' authoring system comprising a page display management system, text and graphics editor and 'exam mode' capacity to store replies and generate run only 'books'. In this respect DOMINO should be compared with other systems like XCAL and VIDEO-SLIDE, it is, however, much more powerful than either of these - especially with respect to its graphics capabilities. The system will support the use of a light pen, a mouse or a high resolution digitiser pad. Very powerful colour graphics facilities are available (including a pixel level editor).
- EASY** This system is page (frame) based with each page created 'on-screen' through a series of prompts and simple keyboard interactions. Plain language prompts are used to guide the courseware author through the authoring process, no programming knowledge is required. There are facilities for generating text and graphics and it is easy to incorporate video material and external applications programs into a training sequence. A variety of 'special effects' can also be produced such as word animation, data entry simulation, use of screen fields and so on.
- EASYWRITER** EASYWRITER is written BASIC and is prompt driven - the user being guided step by step through frame and lesson creation by the computer. AS EASYWRITER lesson is created as a sequence of linked frames, The BASIC program that is responsible for implementing the frame/lesson specification dialogue with the user also generates the lesson. This is then stored as a series of string arrays in a dialect to the PILOT author language. Those are subsequently interpreted (by the PILOT interpreter) when the (R)un option from the command menu is selected.
- ENGOL** Like AGRAPH, the ENGOL system is one which is orientated towards the use of graphics as an instructional resource. It is a command structured language that has been primarily created to suit the needs of students of engineering, graphics, drawing, and design. One of the main features of the system is that it lets the student explore the use of a variety of different graphical projections to represent a scene: the student is therefore free to experiment by changing parameters and, in doing so, generate new views. ENGOL is an acronym for Engineering Graphics Orientated Language.
- FOIL** FOIL (Foil Orientated Interpretive Language) was developed in order to provide a conversational lesson writing facility for instructional programmers who had

access to a general purpose, time-sharing computer system. Programs written in FOIL were prepared using an editor and stored in direct access files which were processed by an interpreter. Like many other early CAL interpreters this was written in FORTRAN. The FOIL authoring system is no longer actively used.

FORTRAN

Of the many languages used for programming a computer, FORTRAN is probably one of the oldest and perhaps one of the most well established - certainly for scientific and engineering applications. There are numbers of examples of its use for developing other authoring systems for instance, the STAF author language is coded in FORTRAN, similarly the BOOK system is also embedded within FORTRAN as a host language.

GPILOT

This is a version of the PILOT author language. It is particularly well suited for implementing CAL/CBT applications requiring graphics support either in the form of static or animated pictures. GPILOT is a contraction of Graphical Programmed Inquiry Learning of Teaching. It is an integral part of the ARTPLAN system.

HAL(1)

A Hypothetical Author Language (HAL) which is assumed to have available within its definition all the facilities provided by a typical author language. A tool of this sort is useful because it enables courseware to be written in a machine/system independent way. HAL could therefore be used as a specification facility in a way which is analogous to the use of pseudo-code in conventional programming language. Although HAL contains some primitive graphics commands, it does not offer any basic control mechanisms for the use of an ancillary instructional resources such as audio or video devices. Needless to say, these could be easily incorporated into HAL. Another weakness of the HAL approach is that it is based on a simple keyboard/display model of student interaction rather than on the more sophisticated workstation model.

HAL (2)

Honeywells Author Language (HAL) was in use during the 1960's and early 1970's/ This approach to CAL authoring is typical of that used by many other CBT systems developed during the 1960's - such as COURSEWRITER and early versions of the TUTOR language.

IIAS

IIAS represents an authoring facility enhancement for use within the general environment provided by IBM's interactive instructional systems. IBMS interactive instructional systems may be broadly divided into two

- basic parts. IIPS Interactive Instructional Presentation System: IIAS Interactive Instructional Authoring System/ IIPS is used to present previous prepared courses. IIAS is used to author Instructional material.
- IPS** the Instructional Programming System attempts to combine the best features of CAI/CBT author languages such as COURSEWRITER and TUTOR with these other facilities only found in high-level programming languages.
- IVL** IVL is an acronym for Interactive Video Learning - a concept that combines video disk (and or video tape) and microcomputer technologies to provide a new form of training/teaching facility. IVL is also the name of one of the many different authoring systems that have been developed for use in this area. The IVL system is an English language prompt-driven authoring facility developed by Dalroth Computer Products. It enables non-computer staff to design, write, and produce interactive video courses using a computer keyboard.
- LISP** LISP is a functional, symbolic, recursive, logical list-processing language that was developed in the early 1960's by John McCarthy of MIT. The language is often associated with various applications of artificial intelligence. LISP is used in a number of CAL/CBT application areas - particularly in the context of knowledge-based CAL.
- LOGO** It is a recursive, procedure - based language designed for educational applications of the computer. Its main use has been for teaching mathematical concepts, heuristics, and formal methods in courses ranging from the primary level through to university.
- LYRIC** LYRIC is an abbreviation for 'Language for Your Remote Instruction by Computer' This was a machine - independent language for a Cal system known as MAESTRO (Method assisted Educational System for Teaching by Remote Operation).
- MASTWRITER** This is an authoring system designed by MAST Learning systems for use in conjunction with training applications involving interactive audio. Lessons produced by this system can also utilise other media - such as a slide projection system.
- MCGRAW-HILL (IAS)** The McGraw Hill Interactive Authoring System has been designed primarily for use on a suitably configured IBM PC. The system consists of two basic parts. an authoring facility and a presentation facility. The authoring system is used to initialise lesson disks

and to create lessons consisting of text, questions, graphics and segments of video. The presentation system uses the previously prepared lesson disks to deliver instructional material to the student/trainer. Six basic types of screen can be created: presentation, multiple choice, matching, fill-in-the-blank, application-simulation and video. Student replies can be validated in a variety of different ways: numerical answers, answer ranges, individual choices and constructed responses. A user exit feature is also available; this allows the creation of lessons that have break-points at which students can load and interact with other IBM application software.

MENTOR

This is a block structured programming language that was developed by Bolt Beconek, and Newman in the USA during the 1960's. It is claimed to have facilities that enable the programmer to model the students knowledge of particular subjects.

MENTOR (2)

Distributed by the Provident Financial Group PCL, MENTOR is a CBT courseware development and delivery facility for use on microcomputers. The system allows the micro to link to the video equipment and slide projectors - thereby providing access to course material that may already be available on these media. The system can use a number of different student interaction devices provided that the microcomputer used as the delivery station is itself able to support such peripherals. Lessons may be based upon conventional CRT instruction or may also involve the use of aptitude tests, business games and simulations.

MICROTEXT

Originating from the National Physical Laboratory in the UK, MICROTEXT is a CRT frame-based dialogue programming language that can be used as an effective means of producing computer-assisted learning materials for use on a microcomputer.

MRC

MRC is a contraction of MrComputer - a simple language for implementing computer-assisted instruction. It was developed by the General Electric Research and Development Centre for in-house CBT applications

MUMEDALA

MUMEDALA is an acronym for MULTi-MEDia Authoring Language. This was an experimental authoring facility developed by Darker, Skipper and Singh. It was designed to be used as a vehicle for exploring the problems encountered during the development of courseware for use within the complex

multi-media interaction environments created by sophisticated CAL/CBT workstations.

- NATAL** In the early 1970's the National Research Council of Canada commissioned the development of a NATional Author Language called NATAL. In laying down the specifications for this language, four basic design goals were considered to be important: ease of use; language portability; terminal independence; and the provision of powerful computational, logical and file-handling facilities.
- PASCAL** Several commercially available courseware development tools employ Pascal as their base programming language. The COMBAT systems uses UESP-pascal in this way.
- PASS** PASS is an acronym for Professional Authoring Software System. It is distributed by audio-visual specialists Bell and Howell. The system is designed to run on a micro-computer, facilities exist for controlling video tape, video disk, and a variety of other image projection and sound-projection equipment; PASS provides a special graphics editor to facilitate the provision of pictorial support for a CBT lesson. A major limitation of the PASS system is its lack of computational facilities.
- PHILVAS** This is an interactive video authoring language produced by Philips, the basic system (which runs on a microcomputer) provided facilities for preparing control programs for an optical laser disk player. Information retrieved from the video disk can be combined with overlay information retrieved from the video disk can be combined with overlay information that has been previously prepared and stored in a suitable storage medium.
- PHOENIX** The PHOENIX system is a CBT production and development facility developed by Goal Systems International in the USA. It consists of two basic parts: a Presentation System (for the delivery of the CBT lessons); and an author language called EASE (for courseware production).
- PILOT** PILOT is an acronym for PROgramed Inquiry Learning or Teaching. A number of dialects of the language exist.
- PLANIT** PLANIT (an acronym for Programming LANguage for Interaction and Teaching) was developed in the late 1960's

PROLOG	Prolog is a language that is well suited for computer application involving symbolic computation and logic programming. The language is conversational and is intended for use with an interactive time-shared computer or microcomputer system.
SAL	SAL; an acronym for Simple Author Language is an example of a multi-media authoring system that was designed for use on a microcomputer system.
SMALLTALK	SMALLTALK is often described as an object-orientated programming system. the basic philosophy behind the language is one of 'communicating objects'. The programmer is able to define and create many different types of object, all of which can communicate with each other by message passing. The language is very suitable for rapid prototyping of ideas, it has thus formed the basis for a number of novel approaches (such as 'programming by rehearsal' and 'application kits') in courseware development.
SMART	SMART is an acronym for System for Managing Authors, Resources, and Training. It operates as an integral part of the WISE CAL/CBT system to provide a complete instructional development, delivery and management system allowing individualised instruction - regardless of the number of students involved, a number of causes or objectives, media mix, testing requirements, or personal scheduling requirements. The SMART system can automatically provide student progress records, curriculum presentations, resource utilisation reports, and a variety of other management information. SMART is menu driver (using natural language prompts) and can invoke a variety of different student testing and assessment methods; performance data can be stored in computer files and can be subsequently accessed in order to generate a variety of different types of report.
STAF	The Science Teachers Authoring Facility (STAF) is an author language that was developed during the 1970's; in order to support the courseware development activity of a large UK teaching project called CALCHEM. Although the system, was designed primarily used by chemists, its facilities are also very applicable to the development of CAL material for other subject areas. Although the language was developed primarily for tutorial purposes it can be used for simulation and modeling techniques within the tutorial dialogue if this is desired.
TenCORE	TenCORE is an author facility that has been designed for use on an IBM PC. Some of the attractive features

of the TenCORE system include an English-like authoring language, integrated source code, graphics and character editor, complete computational abilities; many automatic build in CAL/CBT functions including extensive input student processing (such as spelling checking); and instantaneous switching between authoring and student modes. The characters set editor allows the creation of author defined characters in variable grid sizes. Similarly the graphics editor allows a mouse. TenCORE also provides an optional computer-management instruction system that is able to provide complete student record keeping and test generation facilities as well as the ability to create individual student study prescriptions based upon performance.

TEST

This system was originally developed for use in conjunction with the teaching of language skills using interactive audio techniques. However, it could be employed within virtually any CAL/CBT situation that requires audio support. The major drawback of the system is the limited range of micro-computers for which it is available.

TICCIT

TICCIT is an acronym for Time-shared Interactive Computer Controlled Information Television. It was a micro-computer based CAL system that was designed during the 1970's to support a number of simultaneous learners. Authors could develop courseware using a special editor called APT (Authoring Procedure for TICCIT) in which was embedded proven lesson strategies. For those who wished to program their own pedagogic strategies into their courseware a conventional author language called TAL (TICCIT Author Language) was available. A microcomputer based version Micro TICCIT is also available.

Top Class

Top class is essentially a computer aided tutorial/presentation for use on an IBM PC or compatible hardware. The system is composed of a suite of programs (CREATE, IMAGE, EDITOR and so on) that provide a comprehensive range of screen preparation and presentation facilities. CRT presentations can be text or text combined with graphics. Symbols can be made to move - thereby providing simple animation effects. Sound can be added simply by specifying the note to be played and its duration.

TUTOR

This is the author language that was developed for use with the famous PLATO computer based learning system of the University of Illinois in the USA. It was designed for use on a control time-sharing mainframe

- computer system that supported a large number of purpose-built multimedia teaching terminals. One of the outstanding features of the TUTOR language is the capacity it offers for handling interactive graphical images
- USE** USE is the author language facility employed by the regency microcomputer-based CBT system. With its USE author language, though screen, special keyboard and graphics editor the regency CBT system appears very much like a scaled down (single user) version of PLATO
- VIDEO-SLIDE** This system is essentially a CRT frame production and presentation system. The system is very useful for setting up electronic message board systems and instructional aids that require minimal student interaction.
- WISE** WISE is an abbreviation for WICAT Interactive System for Education. Use in conjunction with SMART this system provides an advanced but easy to use authoring and delivery system. WISE is essentially a natural language menu-driven authoring system that eliminates most of the difficult/time-consuming functions normally associated with courseware development. The system also offers a multi-user capability. facilities are available to enable complex branching and the use of simulations. Time limits can be placed on students responses which may be made via a keyboard or a touch screen. A large repertoire of built in routines are available for judging free response replies - for example, synonym dictionaries, ignorable words, keywords, spelling tolerance algorithms, and numeric ranges. Prescriptive criteria are also available for weighing and scoring student responses.
- XCAL** XCAL is a simple low cost microcomputer based authoring system that is in many ways simulate to the VIDEO-SLIDE system.

Appendix C Assignment 1 Year 1

C.1 Assignment Question Sheet

43.406 Technological Mathematics 11

Assignment 1

Due: 15 May

Background:

You have been asked to develop some working process capability guidelines for a new machine. The machine produces a small pin for use in expensive electronic equipment, which has a critical variable quality characteristic. The precision to which you are able to produce these is the reason your company has the contract. Your boss has charged you with assessing the capability of the new machine.

A first trial run of the machine has been carried out. 100 items were produced, the first 40 have been discarded due to setup errors. The remaining 60 pins have been measured using a calibrated measuring instrument, having sufficient resolution and known accuracy. This is the variables data on which you will be working.

Using Minitab

This assignment requires the use of Minitab Release 8 throughout. Your Minitab course notes "How to use Minitab" will be essential. Minitab Release 8 is available on computers in several computer rooms on the campus.

After starting Minitab, you must generate the data for the assignment. Execute the macro `trnl.mtb` which is on the H: drive. This macro will request the last six digits of your Student ID number, and generate a unique data set for you. The same data will always be generated by the same ID number. As well as the data, the nominal value and specification limits will also be generated for you.

All the required answers should be saved *exactly as specified* in a Minitab worksheet which you must create, with a file name the same as your ID number, e.g. if ID number is 87936281 then your file would be 87936281.mtw. Adhere carefully to, the requirements when placing your answer into the final worksheet, and *do not* add any extra data, notes or titles, etc. Submit this file as part of your assignment along with the graphs requested.

You should submit your Minitab worksheet file on an IBM PC disk, marked with your name and Student ID. The disk will be returned with a file added called `answers.mtw`, which will be the generated answers for your particular data set.

The marks allocated for each part of every question are shown in brackets.

Q1.

- (a) Find the mean and standard deviation of the 60 samples, placing the results in c20 of your answer worksheet.

c2O cell 1 = mean, c2O cell 2 = SD (i.e. c2O(1) and c2O(2) = (y) (2)

- (b) Check that the data can be treated as normal. Obtain a correlation between your original data values and the normal probability values associated with them (store in c2O(3)). (3)
Obtain a normal probability plot, and submit this with your assignment. (1)
- (c) Calculate any possible skew in the data, store the result in c2O(4). (3)
- (d) Calculate any kurtosis in the data, store the result in c2O(5). (3)
- (e) From the following list of comments select the one that best describes your data. Enter the code number for the comment in c2O(6). (2)
1. Data can be considered normal.
 2. Data is non-normal due to skew.
 3. Data is non-normal due to kurtosis.
 4. Data is non-normal due to both skew and kurtosis.'

Q2.

- (a) Calculate the Process Capability Indices for the manufacturing process. The selected nominal value and specification limits are in appropriately labelled columns in your data sheet. Place Cp in c21(1), Cpk in c21(2), Cpm in c21(3), and Cpmk in c21(4). (8)
- (b) From the following list of comments select the one that best describes the capability of your process. Enter the code number for the comment in c21(5). (2)
1. The process is potentially capable.
 2. The process is incapable.
 3. The process is capable.
 4. The process is incapable but centered.

Q3.

- Produce both \bar{x} -bar and R control charts from your data. Record the Upper and Lower control limits in c22. For the \bar{x} -bar chart UCL in c22(1), LCL in c22(2). For the R chart UCL in c22(3), LCL in c22(4) (4)
Submit the plotted charts with your assignment. (2)

C.2 Macro Files

Data generation macro

File: ASS1.MTB

Written for minitab version 8.2 for DOS.

(NB: # means comment line)

noecho

Macro to generate data for assignment one 43.306 Tech Maths 2

```

#
# Written by : V R Breen
# For : J D Tannock
# Date written : 5/4/95
# Last modified : 5/4/95
#
# Data generated by random number generator seeded by the last 6 digits
# of each students ID code
#
# Q1. Normal data: generate 60 data points with mean = 50 and SD = 0.05
# Store in C1
note Please enter the last 6 digits of you ID number
set file 'terminal' c100;
nobs = 1.
let k1 = c100
erase c100
base = k1
random 60 c1;
norm 50 0.05.
# name cells
name c1 = 'Data'
name c2 = 'Nominal'
name c3 = 'USL'
name c4 = 'LSL'
# Q2.
# set the specification limits to be +2.75SD and -3.25SD
let c2(1) = 50
let c3(1) = 50.105
let c4(1) = 49.75
# Q3 requires no extra data values.
#
# Answers to the questions are to be put into c100 onwards.
# Macro end
echo

```

Answer generation macro

File: ANS1.MTB

Written for minitab 8.2 for DOS

A compilation of the generation files and those written to generate the answers for each student. File was then saved under the students code representing numerically their place in the class when listed alphabetically. Individual answer files are aq1.mtb and aq2.mtb.

noecho

Macro to generate data for assignment one 43.306 Tech Maths 2

#

Written by : V R Breen

For : J D Tannock

Date written : 5/4/95

Last modified : 5/4/95

```

#
# Data generated by random number generator seeded by the last 6 digits
# of each students ID code
#
# Q1. Normal data: generate 60 data points with mean = 50 and SD = 0.05
# Store in C1
note Please enter the last 6 digits of you ID number
set file 'terminal' c100;
nobs = 1.
let k1 = c100
erase c100
base = k1
random 60 c1;
norm 50 0.05.
# name cells
name c1 = 'Data'
name c2 = 'Nominal'
name c3 = 'USL'
name c4 = 'LSL'
# Q2.
# set the specification limits to be +2.75SD and -3.25SD
let c2(1) = 50
let c3(1) = 50.105
let c4(1) = 49.75
# Q3 requires no extra data values.
#
# Answers to the questions are to be put into c100 onwards.
#
# Macro to answer Q1. of assignment1 43.306, 1995
# written : 5/4/95
# revised : 4/5/95
# Answers stored in c20
# c20(1) = mean
# c20(2) = SD
# c19 = nscores
# c20(3) = correlation between raw data and nscores
# c20(4) = skewness
# c20(5) = kurtosis
# c20(6) = coded comment.
#
# c30 & c31 used for temp storage
#
# name answer cells
name c20 = 'Q1'
name c19 = 'nscores'
# calculate answers
# mead and standard deviation
let c20(1) = Mean('Data')
let c20(2) = StDev('Data')
# Z values and their correlation to raw data
Nscores 'Data' 'nscores'
name m1 = 'corr1'
correlation 'Data' 'nscores' 'corr1'

```

```

copy 'corr1' c30-c31;
omit 1.
let c20(3) = c30(1)
erase c30-c31
# skewness
let c20(4) = ((1/(count('Data')))*(sum(((('Data' - (c20(1)))/(c20(2))))**3)))
# kurtosis
let c20(5) = ((1/(count('Data')))*(sum(((('Data' - (c20(1)))/(c20(2))))**4))-3)
#store comment in c20(6).
# if statements to work out comments.
let c30(4) = (((c20(5) < -1.5) or (c20(5) > 1.5)) and ((c20(4) > 0.5) or (c20(4) < -
0.5)))
let c30(3) = ((c20(5) < -1.5) or (c20(5) > 1.5))
let c30(2) = ((c20(4) > 0.5) or (c20(4) < -0.5))
let c30(1) = ((c20(5) > -1.5) and (c20(5) < 1.5) and (c20(4) < 0.5) and (c20(4) > -0.5))
let c30(4) = c30(4) *4
let c30(3) = c30(3) *3
let c30(2) = c30(2) *2
let c20(6) = sum(c30)
erase c30
#
# Answers to Q2.
# written : 6/4/95
# revised : 4/5/95
#
# mean = c20(1)  calculated in Q1.
# SD = c20(2)  calculated in Q1.
# N= c2(1)
# USL = c3(1)
# LSL = c4(1)
# c31 & c31 used for temp storage
# Cp in c21(1)
# Cpk in c21(20)
# Cpm in c21(3)
# Cpmk in c21(4)
# comment in c21(5)
#
name c21= 'Q2'
# calculate Cp
let c21(1) = ((c3(1) - c4(1))/(6* c20(2)))
# calculate Cpk
let c30(1) = (c3(1) - c20(1))
let c30(2) = (c20(1) - c4(1))
let c21(2) = ((min(c30))/(3*c20(2)))
#calculate Cpm
let c21(3) = (((c3(1)-c4(1))/2)/(3*(sqrt (c20(2)**2)+((c20(1)-c2(1))**2))))
#calculate Cpmk
let c21(4) = ((min(c30))/(3*(sqrt((c20(2)**2)+((c20(1)-c2(1))**2))))
# comments about data
let c30(1) = ((c21(1) > 1) and (c21(2) < 1))
let c30(2) = ((c21(1) < 1) and (c21(2) < 1))
let c30(3) = ((c21(1) > 1) and (c21(2) > 1))
let c30(4) = ((c21(1) < 1) and (c21(2) > 1))

```

```

let c30(2) = c30(2)*2
let c30(3) = c30(3)*3
let c30(4) = c30(4)*4
let c21(5) = sum(c30)
erase c30
# end of macro
echo

```

Marking macro

File: MRK2.MAC

Run under minitab version 9 for windows. Comparison with master answers, and if necessary recalculation using students earlier figures (catering for roll on effect of errors).

GMACRO

```

mrk
# Global macro to mark 43.306 assignment 1
# Version 9.1
#
# First written : 13/5/95
# last updated : 22/5/95
#
#
noecho
# Enter students code number
#
note Enter students code number (enter it twice on the same line separated by a
single space)
read c999 c997;
file 'terminal';
format (A3,1X,F3);
nobs = 1.
let k21 = c997(1) + 100
#
# Get master answers into correct worksheet to mark
#
read c99-c287;
file 'c:\excel\data\masters.txt';
tab.
#
# Compare the master answers with the students answers
#
let k1 = 0
# set comments file
outfile = 'c:\mtbwin\data\43306\commnet'
# Question 1
# mean
if (c20(1) > (ck21(1) - 0.0001)) and (c20(1) < (ck21(1) + 0.0001))
let k1 = k1 + 1
note Question 1, mean - correct
endif

```

```

# std dev
if (c20(2) > (ck21(2) - 0.0001)) and (c20(2) < (ck21(2) + 0.0001))
let k1 = k1 + 1
note Question 1, standard deviation - correct
endif
# correlation
if (c20(3) > (ck21(3) - 0.0001)) and (c20(3) < (ck21(3) + 0.0001))
let k1 = k1 + 3
note Question 1, correlation - correct
endif
# skew
if (c20(4) > (ck21(4) - 0.001)) and (c20(4) < (ck21(4) + 0.001))
let k1 = k1 + 3
note Question 1, skew - correct
else
let c20(10) = ((1/(count('Data')))*(sum(((('Data' - (c20(1)))/(c20(2))))**3)))
if (c20(4) > (ck21(10) - 0.0004)) and (c20(4) < (ck21(10) + 0.0004))
let k1 = k1 + 3
note Question 1, skew - correct using you figures
endif
endif
# Kurtosis
if (c20(5) > (ck21(5) - 0.001)) and (c20(5) < (ck21(5) + 0.001))
let k1 = K1 + 3
note Question 1, kurtosis - correct
else
let c20(11) = ((1/(count('Data'))*(sum(((('Data' - (c20(1)))/(c20(2))))**4))-3)
if (c20(5) > (ck21(11) - 0.0004)) and (c20(5) < (ck21(11) + 0.0004))
let k1 = k1 + 3
note Question 1, kurtosis - correct using your figures
endif
endif
# comment
if (((c20(4) > -0.5) and (c20(4) < 0.5) and (c20(5) > -1.5) and (c20(5) < 1.5)) and
(c20(6) = 1))
let k1 = k1 + 2
note Question 1, comment - correct using your figures
elseif ((c20(4) > -0.5) and (c20(4) < 0.5) and (c20(6) = 3))
let k1 = k1 + 2
note Question 1, comment - correct using your figures
elseif ((c20(5) > -1.5) and (c20(5) < 1.5)) and (c20(6) = 2))
let k1 = k1 + 2
note Question 1, comment - correct using your figures
elseif (c20(6) = 4)
let k1 = k1 + 2
note Question 1, comment - correct using your figures
endif
# Question 2
# Preparation for modified marking
let c995(1) = (c3(1) - c20(1))
let c995(2) = (c20(1) - c4(1))
#
# Cp

```

```

if (c21(1) > (ck21(7) - 0.001)) and (c21(1) < (ck21(7) + 0.001))
let k1 = k1 + 2
note Question 2, Cp - correct
else
let c21(10) = ((c3(1) - c4(1))/(6*c20(2)))
if (c21(1) > (ck21(10) - 0.0004)) and (c21(1) < (ck21(10) + 0.0004))
let k1 = k1 + 2
note Question 2, Cp - correct using you figures
endif
endif
# Cpk
if (c21(2) > (ck21(8) - 0.001)) and (c21(2) < (ck21(8) + 0.001))
let k1 = k1 + 2
note Question 2, Cpk - correct
else
let c21(11) = ((min(c995))/(3*c20(2)))
if (c21(2) > (ck21(11) - 0.0004)) and (c21(2) < (ck21(11) + 0.0004))
let k1 = k1 + 2
note Question 2, Cpk - correct using your figures
endif
endif
# Cpm
if (c21(3) > (ck21(9) - 0.001)) and (c21(3) < (ck21(9) + 0.001))
let k1 = k1 + 2
note Question 2, Cpm - correct
else
let c21(12) = (((c3(1) - c4(1))/2)/((sqrt((c20(2)**2) + ((c20(1) - c2(1))**2)))*3))
if (c21(3) > (ck21(12) - 0.0004)) and (c21(3) < (ck21(12) + 0.0004))
let k1 = k1 + 2
note Question 2, Cpm - correct using your figures
endif
endif
# Cpmk
if (c21(4) > (ck21(10) - 0.001)) and (c21(4) < (ck21(10) + 0.001))
let k1 = k1 + 2
note Question 2, Cpmk - correct
else
let c21(13) = ((min(c995))/(sqrt((c20(2)**2) + ((c20(1) - c2(1))**2)))*3))
if (c21(4) > (ck21(13) - 0.0004)) and (c21(4) < (ck21(13) + 0.0004))
let k1 = k1 + 2
note Question 2, Cpmk - correct using your figures
endif
endif
# Comment
if ((c21(1) > 1) and (c21(2) > 1) and (c21(5) = 3))
let k1 = k1 + 2
note Question 2, comment - correct using your figures
if ((c21(1) > 1) and (c21(5) = 1))
let k1 = k1 + 2
note Question 2, comment - correct using your figures
if ((c21(2) > 1) and (c21(5) = 4))
let k1 = k1 + 2
note Question 2, comment - correct using your figures

```

```

if (c21(5) = 2)
let k1 = k1 +2
note Question 2, comment - correct using your figures
endif
# Question 3
# X-bar UCL
if c22(1) = ck21(12)
let k1 = k1 + 1
note Question 3, X-bar UCL - correct
endif
# X-bar LCL
if c22(2) = ck21(13)
let k1 = k1 + 1
note Question 3, X-bar LCL - correct
endif
# R UCL
if c22(3) = ck21(14)
let k1 = k1 + 1
note Question 3, R-bar UCL - correct
endif
# R LCL
if c22(4) = ck21(15)
let k1 = k1 + 1
note Question 3, X-bar UCL - correct
endif
nooutfile
# Output final mark (graph marks to be added by user)
print k1
print k21
#
read c998;
file 'c:\mtbwin\data\43306\files.txt';
format (A30).
write 'c:\mtbwin\data\43306\port.bat' c998-c999
system c:\mtbwin\data\43306\port.bat
# Copy of answers to students disk is done by a batch file created each time
#
echo
ENDMACRO

```

File transfer batch files

File: PORT.BAT

Executed by mrk2.mac

Number refers to the students alphabetical placement in the class list

```

c:\mtbwin\data\43306\trans.bat 005
005

```

File: TRANS.BAT

Executed by port.bat

Copy the appropriate student file onto their disk along with the comments file generated when marking. These files are then deleted to be recreated again for the next student.

```
copy c:\mtbwin\data\43306\%1.mtw a:\answers.mtw
```

```
copy c:\mtbwin\data\43306\comment.lis a:
```

```
del c:\mtbwin\data\43306\comment.lis
```

```
del c:\mtbwin\data\43306\port.bat
```

Other associated files

No other files are necessary as each student has individual data throughout the assignment. The students individual answer files are also in the assignment archive.

Appendix D Experimental Design Assignment - year one

D.1 Assignment Question Sheet

43.306 *Technological Mathematics*

Assignment 3

Due Wednesday 18th October

Total Marks 80

Q1. You have been contracted to advise the local hospital board on the impression of customer satisfaction as it relates to clinic waiting times. Listed is the waiting times collected for five of the hospitals clinics. The manager wishes to know if there is a difference in the waiting times so that improvements can be made.

The data for this question will be placed in c 1 -c3 (and labelled) upon execution of the macro tm2a2gen.mac. Type **%H:\tm2a2gen** at the Minitab session prompt.

- (a) Analyse the data using a Tukey test with an family error rate of 1%. Enter the mean and standard deviation of the waiting times in the table starting at C25 in the worksheet. Identify any significant differences in the waiting times and enter them in the table starting at C30 in the worksheet.

10 Marks

- (b) Print a boxplot the data for all five clinics on a single graph.

5 Marks

- (c) Perform a multiple comparison (using "with the Best" option) at a family error rate of 2%, after the removal of any data columns that do not match the assumptions for a valid ANOVA test. Place the intervals for level means in the table starting at C40 in the worksheet.

10 Marks

Save the worksheet as TM2Q1.MTW onto a floppy disk.

Q2. You have been asked to continue with the analysis work of a colleague while they are away. The process itself has been under study for some time and an initial model has been developed. The values of the two critical independent variables in the process along with the yield for the last production run are listed in the worksheet called Q2RAW.MTW, (this is on the H:\ASSN2 directory). The manager wishes to know how he can predict the yield of the process using these variables. It is suspected that the model may be non-linear in some factors.

- (a) From the initial prediction model plot and print the residuals of the initial model. Refine the model further so that the manager can be sure that this model takes into account at least 97% of the variation of the recorded data. Place the coefficient terms

in C25 of the worksheet following the pattern $Y = ax_1 + bx_2 + cx_1^2 + dx_2^2 + ex, \dots + \text{constant}$, where the constant is stored in row 1, a in row 2, b in row 3, c in row 4, etc.

13 Marks

- (b) Place in C26 of the work-sheet the 95% CI's for the coefficients (using $t_{\alpha/2, v} = 2.03$).

12 Marks

Save the worksheet as TM2Q2.MTW onto a floppy disk.

- Q3. In the file Q3RAW.MTW (in the H.\.issn2 directory) you will find the results of a number of initial modeling exercises. Determine for each set of residuals (labelled RESII, RES12, RES13) what is missing from the existing model. Enter into C25 as labelled in C24 the number of the problem from the following list of possible reasons.

1. No constant present
2. Squared term required
3. Cross product term required
4. Cubic term required
5. Time or sequence dependency in the residuals
6. Heteroscedasticity present in residuals

Save the worksheet as TW2Q3.MTW onto a floppy disk. -

15 Marks

- Q4. Construct a design matrix for a factorial experiment that has 7 factors. **Do not randomise the matrix.** There is a critical material that must be present in all of the tests to be carried out. Each run will require 5g of this material and only 313g are available to be used in the first set of experiments. It is intended that all of the material is used in the experiment. The matrix should be generated with the following levels for each term

Low	High
100	200
59	35
-2	0
35	99
12	50
-10	-5
0	1.5

Save the matrix in C1-C7. Type `%H:\TM2Q4YS` at the Minitab session prompt to generate the resultant yields from your design. Fit a factorial model to your design using all the terms in the model. Store the effects of this model (these should be in C9).

Save the worksheet as TW2Q4.MTW onto a floppy disk.

15 Marks

To be submitted:

- 1 floppy disk containing 4 files, labelled correctly.

- Label the disk with you name and ID number.
 1 print out of a graph of box plots (from Q1)
 1 print of residuals (from Q2)

D.2 Macro Files

Individual data was used for question one only. Questions 2,3,4 were the same for all students and a direct comparison was used in marking them.

Data generation macro

File: TM2A2GEN.MAC

Data generation macro. Used in question one. Created in Minitab version 10 for windows.

GMACRO

tm2a2gen

Macro to generate data for assignment three 43.306 Tech Maths 2

#

Written by : V R Breen

For : D J Barnes

Date written : 3/10/95

Last modified : 3/10/95

#

Data generated by random number generator seeded by the last 6 digits

of each students ID code

#

Q1. Normal data: generate 30 data points with varying means and standard deviations

Store in C1 - c5

note Please enter the last 6 digits of you ID number

set file 'terminal' c100;

nobs = 1.

end

let k1 = c100

erase c100

base = k1

random 30 c1;

norm 12 5.0.

random 30 c2;

norm 10 5.0.

random 30 c3;

norm 5 5.0.

random 30 c4;

norm 7.5 5.0.

random 30 c5;

norm 12 15.0.

name cells

```
name c1 = 'Clinic 1'
name c2 = 'Clinic 2'
name c3 = 'Clinic 3'
name c4 = 'Clinic 4'
name c5 = 'Clinic 5'
#
# name and preformat output columns
name c25 = 'means'
name c26 = 'S.D.'
read c24;
format (A12).
Clinic one
Clinic two
Clinic three
Clinic four
Clinic five
end
name c29 = 'Between'
name c30 = 'Clinic_1'
name c31 = 'Clinic_2'
name c32 = 'Clinic_3'
name c33 = 'Clinic_4'
name c34 = 'Clinic_5'
read c29;
format (A12).
Clinic one
Clinic two
Clinic three
Clinic four
Clinic five
end
name c39 = 'Part C'
name c40 = 'Lower'
name c41 = 'Center'
name c42 = 'Upper'
read c24;
format (A12).
Clinic one
Clinic two
Clinic three
Clinic four
Clinic five
end
# end of macro
ENDMACRO
```

File: TM2A4YS.MAC

Generation of "experimental" yields for question 4. Created in Minitab version 10 for windows.

```
GMACRO
tm2a4ys
# Written for 43.306 Technological Mathematics 2
# macro to generate the Y values for Q4 of Assignment three
# Written for : D J Barnes
# Written by : V R Breen
# Date created : 3/10/95
# Last modified : 3/10/95
#
# enter into c8 the proposed Y values
name c8 = 'Y'
set c8;
nobs 32.
35.5673
35.6380
35.6959
33.7950
31.1314
32.5960
35.8433
34.0283
31.2680
40.0451
38.7293
32.4625
32.1568
32.8602
35.1911
31.6132
35.6547
34.4173
33.7269
33.9652
35.1709
35.4131
31.5247
39.9421
37.9718
35.7211
34.4176
35.4898
35.9862
38.2351
37.6554
32.5109
end
# end of macro
ENDMACRO
```

Questions 2, 3 have Minitab data files which have the initial data set up in them.

Marking

Question one done by hand. Answers generated as they would be by the student and compared by sight. Done this way as Minitab did not allow the storage of the output from the required functions.

Session listing showing set up and execution of question one

file: ASN2.LIS

```
MTB > Base 021197.
MTB > Random 30 c1;
SUBC> Normal 12 5.
MTB > name c1 'Level 1'
MTB > name c2 'Level 2'
MTB > name c3 'Level 3'
MTB > name c4 'Level 4'
MTB > name c5 'Level 5'
MTB > Random 30 'Level 2';
SUBC> Normal 10 5.
MTB > Random 30 'Level 3';
SUBC> Normal 5 5.
MTB > Random 30 'Level 4';
SUBC> Normal 7.5 5.
MTB > Random 30 'Level 5';
SUBC> Normal 12 15.
MTB > Stack ('Level 1') ('Level 2') ('Level 3') ('Level 4') ('Level 5') (c8);
SUBC> Subscripts c9.
MTB > Oneway C8 C9;
SUBC> Tukey 1.
```

One-Way Analysis of Variance

Analysis of Variance on C8

Source	DF	SS	MS	F	p
C9	4	3199.3	799.8	13.77	0.000
Error	145	8421.0	58.1		
Total	149	11620.4			

Individual 95% CIs For Mean
Based on Pooled StDev

Level	N	Mean	StDev	-----+-----+-----+-----
1	30	13.041	6.090	(----*---)
2	30	10.908	5.790	(---*----)
3	30	2.494	5.399	(---*----)
4	30	8.716	4.918	(----*---)

5 30 16.243 12.901 (----*----)
 +-----+-----+-----+-----+
 Pooled StDev = 7.621 0.0 6.0 12.0 18.0

Tukey's pairwise comparisons

Family error rate = 0.0100
 Individual error rate = 0.00115

Critical value = 4.69

Intervals for (column level mean) - (row level mean)

	1	2	3	4
2	-4.392 8.659			
3	4.022 17.073	1.888 14.939		
4	-2.200 10.851	-4.334 8.717	-12.747 0.304	
5	-9.727 3.324	-11.861 1.190	-20.275 -7.224	-14.053 -1.002

MTB > Boxplot 'Level 1' 'Level 2' 'Level 3' 'Level 4' 'Level 5';

SUBC> Box;
 SUBC> Type 0;
 SUBC> EColor 1;
 SUBC> Color 0;
 SUBC> EType 1;
 SUBC> ESize 1;
 SUBC> Width 0.5;
 SUBC> Symbol;
 SUBC> Outlier;
 SUBC> Overlay.

MTB > Oneway C8 C9;

SUBC> MCB 2 +1.

One-Way Analysis of Variance

Analysis of Variance on C8

Source	DF	SS	MS	F	p
C9	4	3199.3	799.8	13.77	0.000
Error	145	8421.0	58.1		
Total	149	11620.4			

The regression equation is

$$Y = 11.3 + 25.0 X1 - 10.1 X2 + 10.0 X1^2$$

Predictor	Coef	Stdev	t-ratio	p
Constant	11.328	1.252	9.05	0.000
X1	25.0241	0.0572	437.26	0.000
X2	-10.0523	0.0369	-272.65	0.000
X1^2	9.9998	0.0010	10227.54	0.000

s = 5.032 R-sq = 100.0% R-sq(adj) = 100.0%

Analysis of Variance

SOURCE	DF	SS	MS	F	p
Regression	3	9136570368	3045523456	1.203E+08	0.000
Error	36	912	25		
Total	39	9136571392			

SOURCE	DF	SEQ SS
X1	1	6391112192
X2	1	96960616
X1^2	1	2648496896

Unusual Observations

Obs.	X1	Y	Fit	Stdev.Fit	Residual	St.Resid
3	80.7	67134.9	67141.5	3.3	-6.6	-1.74 X
10	68.0	47566.8	47557.1	2.3	9.7	2.16R
22	38.4	15810.5	15799.7	1.2	10.8	2.21R
38	-33.9	10245.7	10244.9	3.3	0.8	0.21 X

R denotes an obs. with a large st. resid.

X denotes an obs. whose X value gives it large influence.

```
MTB > Name c6 = 'RESI1'
MTB > Regress 'Y' 2 'X1' 'X2' ;
SUBC> Constant;
SUBC> Residuals 'RESI1'.
```

Regression Analysis

The regression equation is

$$Y = - 844 + 511 X1 + 70.5 X2$$

Predictor	Coef	Stdev	t-ratio	p
Constant	-844	2101	-0.40	0.690
X1	510.83	53.67	9.52	0.000
X2	70.48	60.56	1.16	0.252

s = 8461 R-sq = 71.0% R-sq(adj) = 69.4%

Analysis of Variance

SOURCE	DF	SS	MS	F	p
Regression	2	6488073216	3244036608	45.32	0.000
Error	37	2648497664	71581016		
Total	39	9136570368			

SOURCE	DF	SEQ SS
X1	1	6391112192
X2	1	96960616

Unusual Observations

Obs.	X1	Y	Fit	Stdev.Fit	Residual	St.Resid
3	80.7	67135	40475	3405	26660	3.44R
38	-33.9	10246	-15190	3578	25436	3.32R

R denotes an obs. with a large st. resid.

```
MTB > Plot 'RESI1'*'X1';
SUBC> Symbol.
MTB > Save 'C:\MTBWINDATA\43306\AS2Q2.MTW';
SUBC> Replace.
Saving worksheet in file: C:\MTBWINDATA\43306\AS2Q2.MTW
MTB > Plot 'RESI1'*'X2';
SUBC> Symbol.
MTB > BReg 'Y' 'X1' 'X2' 'X1^2';
SUBC> NVars 1 3;
SUBC> Best 2.
```

Best Subsets Regression

Response is Y

Vars	Adj. R-sq	R-sq	C-p	s	X	X ¹	X ²	X ²
1	99.9	99.9	3E+05	474.93	X			
1	70.0	69.2	1E+08	8499.9	X			
2	100.0	100.0	7E+04	225.60	X	X		
2	99.9	99.9	2E+05	361.75	X	X		
3	100.0	100.0	4.0	5.0319	X	X	X	

```
MTB > Save 'C:\MTBWINDATA\43306\AS2Q2.MTW';
SUBC> Replace.
```

Saving worksheet in file: C:\MTBWIN\DATA\43306\AS2Q2.MTW
* NOTE * Existing file replaced.
MTB > Stop.

Other associated files

Raw data files for questions 2 and 3

Q2RAW.MTW

Q3RAW.MTW

Master answers in Minitab worksheet format

A2Q3.MTW

Question 3

AS2Q2.MTW

Question 2

AS2Q4.MTW

Question 4

A4TSET.MTW

Question 4 test setup

Example of some of the graphs required by assignment

BOXQ1.MGF

Box plot (Q1)

RESID1.MGF

Residual plot

Appendix E Experimental Design Assignment - year two

E.1 Assignment Question Sheet

43.306 Assignment 2

Date due : 19 July

Submission of disk to : Box outside Production Technology Office, labeled 43.306 disks. Label your disk with your name and ID number.

General Instructions

In this assignment you will be required to execute a number of macro programs, store answers in specific locations and to present data in the correct way. If you are unable to execute the macro programs please seek help (see Vivienne Breen RC2.10, ext. 5556). The locations into which to place the answers required will be noted in each question along with the name under which the file is to be saved for each question. The formats of the answers should be self explanatory from the question. A basic guide is that most items are numeric, with comments in alpha columns, i.e. text. Unless otherwise stated enter your response in the next available cell location in the appropriate column, often this will be the first. Column locations are noted in their short form. Thus Column 4 cell (or row) 3, becomes C4(3). Always start a new worksheet for each new question.

Section 1 Basic Statistics

1. Samples have been taken of products produced by an automatic machining process. To obtain the sample measurements run the macro SIQL. (Type %h:slql at the MTB> prompt) Test the hypothesis that the process is still producing good products with a mean measure of 15mm for the critical diameter. The sample measurements are of this diameter only. For this test it is accepted that a σ of 0.2mm is adequate. Enter the Z and P values in columns 5 and 6 respectively, as labelled. In column 7 enter a Yes or No to the question "Is the sample mean equal to the critical value, i.e. 15mm, using a 95% level of confidence", column is labelled Comment. Save the worksheet onto a floppy disk as SIQL.MTW.

10 marks

2. Changes have been made in the operation of the machine. Two samples have been taken, one before the machine was altered and one afterwards. You are asked to test if they differ significantly from each other with respect to the critical diameter. Are the means of both samples are the same? Run the macro S 1 Q2 to obtain the second sample of measurements. Store the t and P values in Column 5 and 6 respectively, as labelled. In column 7 (labelled Comment) enter a Yes or No to the question "Has the process change caused a change in the critical diameter of the product, using a 95% level of confidence". Save the worksheet onto floppy disk as S 1 Q2.MTW.

15 marks

Section 2 Design of Experiments

1. An experimenter has enough material to carry out 36 experimental runs and 7 factors to study. Run the macro S2Q1 to obtain the levels for the 7 factors. These will be stored in the column labelled "levels". Generate a suitable table of experimental conditions and store this in c2 to c8. Once the table is in place run the macro S2Q1EXPT to execute the experiment, the results will be found in c10 which is labelled "results". Analyse the results to obtain the main effects for the 7 factors, store these in c11. List the factor identifier, i.e. column label, for any single factors that cannot be separated from an interaction in c12 (labelled aliased). Save the worksheet onto a floppy disk as S2Q1.MTW.

25 marks

2. Your employer requires you to perform an initial investigation, or screening experiment, to identify the factors that require further investigation. There is only enough material available in this first set of experiments to perform 12 runs. Run macro S2Q2 to obtain the levels for the 11 factors. Generate a treatment table and store this in c2 to c12. Once the table is in place run the macro S2Q2EXPT to execute the experiment, the results will be found in c15 which is labelled "results". Generate the single factor effects and so identify the factors that will be used in further testing. Store the effects in c16 and the factors (i.e. column labels) in c17 (labelled factors). Save the worksheet onto a floppy disk as S2Q2.MTW.

25 marks

3. A company wishing to premix cocktails requires that the prescribed mixture be of an optimal level. Design an experiment to establish the best mixture of the components for a cocktail. As some of the ingredients are of a sensitive nature to the company only their code identifier is available to you. There are four particular items in the cocktail that are of interest. These will be labelled 'A', 'B', 'C,' and 'D' respectively. It has been decided upon a lattice design of 3 degrees. Once your design has been generated and stored in c1- c4, label these with the corresponding code identifier, i.e. A, B, C or D. Execute the macro S2Q3 to simulate the experimental process. Fit a linear mixture model to these results. In c10-c13, labelled with the corresponding code identifier, record the part concentration for each component (i.e. its coefficient in the model equation rounded to the nearest whole number). Save the worksheet onto a floppy disk as S2Q3.MTW.

25 marks

E.2 Macro Files

File: S1Q1.MAC

Generation of master answers for Section one Question one.

GMACRO

noecho

macro to generate master answers for 43.306 1996 S1Q1

#

```
# Written by : V R Breen
# Date written : 11/6/96
# Last modified : 11/6/96
#
# Generate data points
note Enter last 6 digits of students ID number
set c4;
file 'terminal';
nobs = 1.
let k1 = c4
random 60 c1;
norm 15 0.2.
# generate function answer
ZTest 15 0.2 C1;
Alternative 0.
# Copy by hand to data blocks
ENDMACRO
```

File: S1Q2.MAC

Data generation for Section one Question two

GMACRO

```
noecho
# Macro to generate data for Section 1 Question 2 43.306 1996
#
# Written by : V R Breen
# For : D J Barnes
# Date written : 16/5/96
# Last modified : 16/5/96
#
# Set base to students ID-Yr digits (ie last 6)
note Please enter the last 6 digits of you ID number
set c100;
file 'terminal';
nobs =1.
let k1 = c100
erase c100
base = k1
# Generate data points
random 60 c1;
norm 15 0.2.
name c1 'Sample 1'
random 60 c2;
norm 20 2.
name c2 'Sample 2'
# name answer columns
name c5 'T'
name c6 'P-value'
name c7 'Comment'
#end of macro
```

ENDMACRO

File: S2Q1.MAC

Data generation for Section two Question one

GMACRO

noecho

Macro to generate data for Section 2 Question 1 43.306 1996

#

Written by : V R Breen

For : D J Barnes

Date written : 16/5/96

Last modified : 16/5/96

#

Set base to students ID-Yr digits (i.e. last 6)

note Please enter the last 6 digits of you ID number

set c100;

file 'terminal';

nobs =1.

let k1 = c100

erase c100

base = k1

Generate levels, 7 factors (c2 - c8)

let k2 = k1 - ((round(k1/10))*10)

name c1 'Levels'

if k2 = 0

set c1;

nobs = 14.

12

14

2

7

5

11

1

19

22

26

19

21

3

9

end

elseif k2 = 1

set c1;

nobs =14.

17

19

6

11

7

```
13
7
25
17
19
24
26
12
18
end
elseif k2 = 2
set c1;
nobs = 14.
9
11
15
20
3
9
17
35
8
10
4
6
2
8
end
elseif k2 = 3
set c1;
nobs = 14.
5
7
2
7
4
10
19
37
3
5
7
9
22
28
end
elseif k2 = 4
set c1;
nobs = 14.
9
11
5
10
```

```
18
24
3
21
12
14
9
11
13
19
end
elseif k2 = -5
set c1;
nobs = 14.
10
12
5
10
9
15
7
25
9
11
7
9
1
7
end
elseif k2 = -4
set c1;
nobs = 14.
4
6
5
10
12
18
6
24
74
76
7
9
2
8
end
elseif k2 = -3
set c1;
nobs = 14.
6
8
7
```

```
12
73
74
7
25
6
8
3
5
4
10
end
elseif k2 = -2
set c1;
nobs = 14.
9
11
10
15
2
8
0
18
76
78
9
11
2
8
end
else
set c1;
nobs = 14.
8
10
3
8
51
57
2
20
95
97
3
5
7
13
end
endif
# name matrix columns
name c2 'A'
name c3 'B'
name c4 'C'
```

```
name c5 'D'
name c6 'E'
name c7 'F'
name c8 'G'
# end of macro
ENDMACRO
```

File: S2Q1EXPT.MAC

Simulation of process execution for Section two Question one

GMACRO

```
noecho
# Macro written to simulate the experimental results for S2Q1 43.306 1996
#
# Written by : V R Breen
# For : D J Barnes
# Date written : 16/5/96
# Last modified : 16/5/96
#
# Generate random error for experimental runs
base = k1
random 32 c100;
norm 0 1.
# Generate experimental results
let c10 = (19*'A')+(2*'B')-(17*'C')+(0.5*'D')+(0.01*'E')-(3*'F')-(0.07*'G')+c100
erase c100
name c10 'Results'
name c11 'Effects'
name c12 'Aliased'
# end of macro
ENDMACRO
```

File: S2Q2.MAC

Data generation for Section two Question two

GMACRO

```
noecho
# Macro to generate data for Section 2 Question 2 43.306 1996
#
# Written by : V R Breen
# For : D J Barnes
# Date written : 16/5/96
# Last modified : 16/5/96
#
# Set base to students ID-Yr digits (ie last 6)
note Please enter the last 6 digits of you ID number
set c100;
```

```
file 'terminal';
nobs =1.
let k1 = c100
erase c100
base = k1
# Generate levels, 7 factors (c2 - c8)
let k2 = k1 - ((round(k1/10))*10)
name c1 'Levels'
if k2 = 0
set c1;
nobs = 14.
12
14
2
7
5
11
1
19
22
26
19
21
3
9
end
elseif k2 = 1
set c1;
nobs =14.
17
19
6
11
7
13
7
25
17
19
24
26
12
18
end
elseif k2 = 2
set c1;
nobs = 14.
9
11
15
20
3
9
```

```
17
35
8
10
4
6
2
8
end
elseif k2 = 3
set c1;
nobs =14.
5
7
2
7
4
10
19
37
3
5
7
9
22
28
end
elseif k2 = 4
set c1;
nobs = 14.
9
11
5
10
18
24
3
21
12
14
9
11
13
19
end
elseif k2 = -5
set c1;
nobs = 14.
10
12
5
10
9
```

```
15
7
25
9
11
7
9
1
7
end
elseif k2 = -4
set c1;
nobs = 14.
4
6
5
10
12
18
6
24
74
76
7
9
2
8
end
elseif k2 = -3
set c1;
nobs = 14.
6
8
7
12
73
74
7
25
6
8
3
5
4
10
end
elseif k2 = -2
set c1;
nobs = 14.
9
11
10
15
```

```
2
8
0
18
76
78
9
11
2
8
end
else
set c1;
nobs = 14.
8
10
3
8
51
57
2
20
95
97
3
5
7
13
end
endif
# name matrix columns
name c2 'A'
name c3 'B'
name c4 'C'
name c5 'D'
name c6 'E'
name c7 'F'
name c8 'G'
# end of macro
ENDMACRO
```

File: S2Q2EXPT.MAC

Simulation of process execution for Section two Question two

GMACRO

noecho

Macro written to simulate the experimental results for S2Q1 43.306 1996

#

Written by : V R Breen

For : D J Barnes

Date written : 16/5/96

```
# Last modified : 16/5/96
#
# Generate random error for experimental runs
base = k1
random 12 c100;
norm 0 1.
# Generate experimental results
let c15 = (12*'A')+(17*'C')+(2*'E')+(19*'F')-(20*'H')+(7*'K')+c100
erase c100
name c15 'Results'
# end of macro
ENDMACRO
```

File: S2Q3.MAC

Simulation of the process for Section two Question three

GMACRO

```
noecho
# Macro written to simulate the experimental results for S2Q3 43.306 1996
#
# Written by : V R Breen
# For : D J Barnes
# Date written : 16/5/96
# Last modified : 16/5/96
#
# Generate random error for experimental runs
# Set base to students ID-Yr digits (ie last 6)
note Please enter the last 6 digits of you ID number
set c100;
file 'terminal';
nobs =1.
let k1 = c100
erase c100
base = k1
random 20 c100;
norm 0 0.001.
# Generate experimental results
let c6 = c1*4 + c2*2 + c3 +2*c4 + c100
name c6 'Results'
erase c100
name c10 'factor A'
name c11 'factor B'
name c12 'factor C'
name c13 'factor D'
# end of macro
ENDMACRO•
```

Assignment Marking**File: MS1Q1.MAC**

Marking of Section one Question one

```
GMACRO
mslq1
#noecho
# marking macro file for 43.306 1996
# macro covers Question 1 of Section 1
#
# written for : D J Barnes
# written by : V R Breen
# date written : 30/7/96
# last modified : 31/7/96
#
#
# open student file and merge in master file to start at c100
#
# Compare Z and P values with master answers according to ID number
note enter students code number
set c99;
file 'terminal';
nobs =1.
let k2 = 0
let k1 = c99
if (c5(1) = c101(k1))
let k2 = k2 + 2
endif
if (c6(1) = c102(k1))
let k2 = k2 + 2
endif
# check comment
set c97;
nobs = 6.
1 1 1 0 0 0
end
read c98;
format (A3).
yes
Yes
YES
no
No
NO
end
Convert C98 C97 'Comment' c8
if (c6(1) >= 0.05 and c8(1) = 1)
let k2 = k2 + 6
elseif (c6(1) < 0.05 and c8(1) = 0)
let k2 = k2 + 6
endif
# report results
print k2
#end of macro
ENDMACRO
```

File: MS1Q2.MAC

Marking of Section one Question two

GMACRO

ms1q2

#noecho

marking macro file for 43.306 1996

macro covers Question 2 of Section 1

#

written for : D J Barnes

written by : V R Breen

date written : 30/7/96

last modified : 31/7/96

#

#

open student file and merge in master file to start at c100

#

Compare Z and P values with master answers according to ID number

note enter students code number

set c99;

file 'terminal';

nobs = 1.

let k2 = 0

let k1 = c99

if (c5(1) = c103(k1))

let k2 = k2 + 3

endif

if (c6(1) = c104(k1))

let k2 = k2 + 3

endif

check comment

set c97;

nobs = 6.

1 1 1 0 0 0

end

read c98;

format (A3).

yes

Yes

YES

no

No

NO

end

Convert C98 C97 'Comment' c8

if (c6(1) > 0.05 and c8(1) = 0)

let k2 = k2 + 9

elseif (c6(1) <= 0.05 and c8(1) = 1)

let k2 = k2 + 9

endif

report marks

print k2

```
# end of macro
ENDMACRO
```

File: MS2Q1.MAC

Marking of Section two Question one

GMACRO

```
ms2q1
#noecho
# marking macro file for 43.306 1996
# macro covers Question 1 of Section 2
#
# written for : D J Barnes
# written by : V R Breen
# date written : 30/7/96
# last modified : 30/7/96
#
#
# open student file and merge in master file to start at c100
#
# Compare Z and P values with master answers according to ID number
note Enter last 6 digits of ID number
set c99;
file 'terminal';
nobs =1.
let k2 = 0
let k1 = c99
base = k1
random 32 c100;
norm 0 1.
# generate design matrix for comparison
FFDesign 7 32;
XMatrix c20-c26;
Levels 'Levels'.
# regenerate results
let c28 = (19*c20)+(2*c21)-(17*c22)+(0.5*c23)+(0.01*c24)-(3*c25)-
(0.07*c26)+c100
# generate effects
Name c30 = 'EFFE10'
FFactorial c28 = c20-c26;
Effects 'EFFE10'.
# compare with student answers
let c35 = (c2 = c20)
let c36 = (c3 = c21)
let c37 = (c4 = c22)
let c38 = (c5 = c23)
let c39 = (c6 = c24)
let c40 = (c7 = c25)
let c41 = (c8 = c26)
let c42 = (c11 = c30)
let c43(1) = sum(c35)
let c43(2) = sum(c36)
```

```

let c43(3) = sum(c37)
let c43(4) = sum(c38)
let c43(5) = sum(c39)
let c43(6) = sum(c40)
let c43(7) = sum(c41)
let c43(8) = sum(c42)
# Allocate marks
let c43(9) = c43(1) + c43(2) + c43(3) + c43(4) +c43(5) +c43(6) +c43(7)
if c43(9) = 224
let k2 = k2 + 10
endif
if c43(8) = 7
let k2 = k2 + 10
endif
# test alliaising
set c50;
nobs = 14.
1 1 2 2 3 3 4 4 5 5 6 6 7 7
end
read c51;
format (A1);
nobs = 14.
a
A
b
B
c
C
d
D
e
E
f
F
g
G
end
Convert C51 C50 c12 c45
if
((c45(1)=3)or(c45(2)=3)or(c45(3)=3)or(c45(4)=3)or(c45(5)=3)or(c45(6)=3)or(c45(7)
)=3))
let k2 = k2 + 1.25
endif
if
((c45(1)=5)or(c45(2)=5)or(c45(3)=5)or(c45(4)=5)or(c45(5)=5)or(c45(6)=5)or(c45(7)
)=5))
let k2 = k2 + 1.25
endif
if
((c45(1)=6)or(c45(2)=6)or(c45(3)=6)or(c45(4)=6)or(c45(5)=6)or(c45(6)=6)or(c45(7)
)=6))
let k2 = k2 + 1.25
endif

```

```

if
((c45(1)=7)or(c45(2)=7)or(c45(3)=7)or(c45(4)=7)or(c45(5)=7)or(c45(6)=7)or(c45(7)
)=7))
let k2 = k2 + 1.25
endif
# display mark
print k2
#end of macro
ENDMACRO

```

File : MS2Q2.MAC

Marking of Section two question two

GMACRO

```

ms2q2
#noecho
# marking macro file for 43.306 1996
# macro covers Question 1 of Section 2
#
# written for : D J Barnes
# written by : V R Breen
# date written : 30/7/96
# last modified : 30/7/96
#
#
# open student file and merge in master file to start at c100
#
# Compare Z and P values with master answers according to ID number
note Enter last 6 digits of ID number
set c99;
file 'terminal';
nobs =1.
let k2 = 0
let k1 = c99
base = k1
random 12 c100;
norm 0 1.
# Generate matrix
PBDesign 11 12;
XMatrix c20-c30;
Levels 'Levels'.
# generate results
let c35 = (12*c20)+(17*c22)+(2*c24)+(19*c25)-(20*c27)+(7*c30)+c100
# generate effects
Name c36 = 'EFFE10'
FFactorial c35 = c20-c30;
Effects 'EFFE10'.
# convert factors for marking
set c70;
nobs = 22.
1 1 2 2 3 3 4 4 5 5 6 6 7 7 8 8 9 9 10 10 11 11

```

```
end
read c71;
format (A1);
nobs = 22.
a
A
b
B
c
C
d
D
e
E
f
F
g
G
h
H
i
I
j
J
k
K
end
Convert C71 C70 c17 c39
# compare student answers
# matrix
let c40 = (c2 = c20)
let c41 = (c3 = c21)
let c42 = (c4 = c22)
let c43 = (c5 = c23)
let c44 = (c6 = c24)
let c45 = (c7 = c25)
let c46 = (c8 = c26)
let c47 = (c9 = c27)
let c48 = (c10 = c28)
let c49 = (c11 = c29)
let c50 = (c12 = c30)
# effects
let c53 = (c16 = c36)
# aggregate matrix comparisons
let c55(1) = sum(c40)
let c55(2) = sum(c41)
let c55(3) = sum(c42)
let c55(4) = sum(c43)
let c55(5) = sum(c44)
let c55(6) = sum(c45)
let c55(7) = sum(c46)
let c55(8) = sum(c47)
let c55(9) = sum(c48)
```

```

let c55(10) = sum(c49)
let c55(11) = sum(c50)
# aggregate effects comparison
let c55(12) = sum(c53)
# allocate marks
# matrix
let c55(13) = c55(1)+c55(2)+c55(3)+c55(4)+c55(5)+c55(6)+c55(7)+
c55(8)+c55(9)+c55(10)+c55(11)
if c55(13) = 132
let k2 = k2 + 5
endif
# effects
if c55(12) = 11
let k2 = k2 + 10
endif
# factors
if (c16(5) >= 10)
if ((c39(1)=1)or(c39(2)=1)or(c39(3)=1)or(c39(4)=1)or(c39(5)=1)or(c39(6)=1))
let k2 = k2 + (5/3)
endif
if ((c39(1)=3)or(c39(2)=3)or(c39(3)=3)or(c39(4)=3)or(c39(5)=3)or(c39(6)=3))
let k2 = k2 +(5/3)
endif
if ((c39(1)=5)or(c39(2)=5)or(c39(3)=5)or(c39(4)=5)or(c39(5)=5)or(c39(6)=5))
let k2 = k2 + (5/3)
endif
if ((c39(1)=6)or(c39(2)=6)or(c39(3)=6)or(c39(4)=6)or(c39(5)=6)or(c39(6)=6))
let k2 = k2 +(5/3)
endif
if ((c39(1)=8)or(c39(2)=8)or(c39(3)=8)or(c39(4)=8)or(c39(5)=8)or(c39(6)=8))
let k2 = k2 + (5/3)
endif
if
((c39(1)=11)or(c39(2)=11)or(c39(3)=11)or(c39(4)=11)or(c39(5)=11)or(c39(6)=11))
let k2 = k2 + (5/3)
endif
elseif (c16(5) < 10)
if ((c39(1)=1)or(c39(2)=1)or(c39(3)=1)or(c39(4)=1)or(c39(5)=1))
let k2 = k2 + 2
endif
if ((c39(1)=3)or(c39(2)=3)or(c39(3)=3)or(c39(4)=3)or(c39(5)=3))
let k2 = k2 + 2
endif
if ((c39(1)=6)or(c39(2)=6)or(c39(3)=6)or(c39(4)=6)or(c39(5)=6))
let k2 = k2 + 2
endif
if ((c39(1)=8)or(c39(2)=8)or(c39(3)=8)or(c39(4)=8)or(c39(5)=8))
let k2 = k2 + 2
endif
if ((c39(1)=11)or(c39(2)=11)or(c39(3)=11)or(c39(4)=11)or(c39(5)=11))
let k2 = k2 + 2
endif
endif
endif

```

```
# report mark
print k2
# end of macro
ENDMACRO
```

File : MS2Q3.MAC

Marking of Section two Question three

GMACRO

```
ms2q3
#noecho
# marking macro file for 43.306 1996
# macro covers Question 1 of Section 2
#
# written for : D J Barnes
# written by : V R Breen
# date written : 31/7/96
# last modified : 31/7/96
#
#
# open student file and merge in master file to start at c100
#
# Compare Z and P values with master answers according to ID number
note Enter last 6 digits of ID number
let k2 = 0
set c99;
file 'terminal';
nobs =1.
let k1 = c99
base = k1
random 20 c100;
norm 0 0.001.
# Generate design matrix for testing
Sl design 4 3;
Total 1;
Xmatrix c20-c23;
Brief 2.
# Generate experimental results
let c25 = c20*4 + c21*2 + c22 +2*c23 + c100
# analyse results and generate coeffs
Name c29 = 'COEF10'
Mixreg c25 = c20-c23;
Brief 2;
Coeff 'COEF10'.
let c30 = round(c29)
# compare student answers
# matrix
let c35 = (c1 = c20)
let c36 = (c2 = c21)
let c37 = (c3 = c22)
let c38 = (c4 = c23)
# aggregate matrix comparisons
```

```
let c40(1) = sum(c35)
let c40(2) = sum(c36)
let c40(3) = sum(c36)
let c40(4) = sum(c38)
# allocate marks
let c40(5) = c40(1) + c40(2) + c40(3) + c40(4)
if c40(5) = 80
let k2 = k2 + 16
endif
if c10(1) = c30(1)
let k2 = k2 + 2.25
endif
if c11(1) = c30(2)
let k2 = k2 + 2.25
endif
if c12(1) = c30(3)
let k2 = k2 + 2.25
endif
if c13(1) = c30(4)
let k2 = k2 + 2.25
endif
# report results
print k2
#endof macro
ENDMACRO
```

Other associated files

Appendix F Design of Experiments and Multivariate Analysis Assignment - year three

F.1 Assignment Sheet

43.328 Assignment 3

Due in : 26 September 1997, at the Production Technology office

In a pelt house the pelts are pickled, graded, packed, pressed and shrink wrapped. The processes involved in pickling the pelts are as follows.

The pelts are transported by a belt conveyor from the wool pulling machine into one of seven Challenge Cook drums. The drums have already been filled with a measured weight of water(WATER1) to which has been added sodium sulphide (NASO1). To this is added a second amount of sodium sulphide (NASO2), mixed with a volume of warm water (WATERV1) and this is pumped into the drum at a rate of (RATE1). The temperature of the warm water is (TEMP1). The length of time of the initial pumping cycle is (TIME1).

Water is then pumped into the drum at a rate (RATE2) until a charge weight of water and pelts (WEIGHT1) as measured by a load cell is reached. Aquaderm A (AQUAD1) and Tetrapol (TETRAP) are added to the drum. This is known as bait, which is used to open up the wool follicles to allow the water to remove the remaining wool.

The drum is then run for a set time (DRUM1) at a speed of (RPM1) revolutions per minute. This is followed by a mixing period of (DRUM2) and a faster speed of (RPM2) to ensure that the Aquaderm A and Tetrapol has mixed properly with the water.

At this point a sulphide sample is then taken. The sample is used to calculate the Mullock Alkalinity Sulphide (ALK1).

The drum is then run for a length period (RECYCLE1) in a recycling mode. A continuous cold water wash is then performed by pumping cold water into the drum at (RATE3) until (VOL1) litres of water have been used. This is done by filling the drum to (FULL1) kilograms total weight, then pumping it down until the load cell reading is (EMPTY1) and this is repeated until (VOL1) litres of water has been used. The waste water is pumped into the effluent system to purify it before it leaves the plant.

Once the cold wash is finished, a warm wash is performed until the temperature of the brine reaches (WBRINET). The washing water temperature which is used to raise the brine temperature is (WARMT1). The drum is pumped down with the waste water going into the effluent again until the load cell read out is (WARMWT). (AMMONIA) kilograms of Ammonia Chloride is added manually.

The pH is then checked after (PHTIME) time. A second charge of Aquaderm A (AQUAD2) is added and the drum is run for (RECYCLE2) on the recycle sequence. Another cold water wash is undertaken where (COLD2) litres of water is pumped into the drum. During this time the drum is continually draining the water away. The volume is calculated by multiplying the Pump rate (CORATE2) multiplied by the time (COTIME2).

(SALT) kilograms of salt is added. After running for a time (SALTIME) (SULPH) litres of sulphuric acid is added into the recycling pipe using a tap and a funnel. This changes the brine into a slightly acid solution. This is when the bleaching of the pelt occurs, which bleaches the pelt a nice white colour.

Busan (BUSAN) is added using a bucket to help preserve the pickled pelt so that it keeps a shiny finish.

Tests are done every (TESTINT) minutes on the salt and acid levels. If the readings are below (MESALT) and (MEASACID) extra acid and salt are added. On the rare occasions where the levels are too acid water is added.

A sample of the finished pelts is taken and the useful area of the pelt is calculated. The ratio of useful area to initial area of pelt is calculated as a **percentage**.

Table 1 summarises the variables identified in the processes and the typical range over which they may be varied.

For the assignment you are to assume that you wish to maximise the ratio of the useable area (expressed as a percentage) of the final pelt to the total area of the initial pelt. Any of the above factors affect this ratio and if an optimum value is to be achieved, it is suspected that there are some non linear effects.

Select a set of factors which you think might be important, and design an experiment using Minitab to determine the effect of these factors. Store the design in the worksheet so that you can use it to calculate their effect. Before running the simulation macro enter the following command at the MTB> prompt in the session window SWAP. Do this for each time you run Minitab, the command need only be entered once per session, but without it the macro will cause an error in Minitab. (See the HINTS section at the bottom of this assignment) Run the macro %H:\exper which will generate an appropriate set of results calculated from the levels you use for your experiments, it will also ask you to enter your ID number. The results will be placed in a column previously labelled 'Results' and is a measure of the percentage increase in useable pelt area. Use the appropriate analysis technique to analyse the results and determine whether the factors you have chosen are significant. Select the significant factors and carry out a further set of experiments until you are clear that you have sufficient data to identify the key factors.

Keep each experimental design in a separate worksheet for each one which should be saved as EXPT1.MTW EXPT2.MTW To EXPTN.MTW where N is the number of experimental designs you use. (Note: Each design will have several treatments).

Use a multi level experiment to determine the relationship between the percentage increase and the factors you have identified. You may need to repeat the experimental process several times to reach this point.

Variable	Variable Name	Lower end of Range	Upper end of Range
Initial charge of water	WATER1	900 kg	1000 kg
Initial charge of Sodium Sulphide	NASO1	20 kg.	30 kg
Second amount of Sodium Sulphide	NASO2	200 kg	300 kg
Volume of water added in first mixing	WATERV1	500 litres	600 litres
Rate of pumping for first mixing	RATE1	180 litres/min	220 litres/min
Temperature of water in first mixing	TEMP1	20 ⁰ C	30 ⁰ C
Length of initial pumping cycle	TIME1	10 min	15 min
First mixing fast pumping speed	RATE2	550 litres/min	650 litres/min
Final weight of water in drum	WEIGHT1	6800 kg	6900 kg
Aquaderm added first time	AQUAD1	950 kg	1000 kg
Tetrapol added	TETRAP	1.2 litres	1.4 litres
Initial drum running time	DRUM1	4 min	6 min
Speed of intial drum mixing	RPM1	3 rpm	5 rpm
Main cycle drum running time	DRUM2	50 min	70 min
Speed of main mixing cycle	RPM2	6 rpm	10 rpm
Mullock Alkalinity Sulphide	ALK1	6	10
Recyling drum operation time	RECYCLE1	2 hours	7 hours
Cold water wash pumping speed	RATE3	500 litres/min	700 litres/min
Volume of water used for wash	VOL1	18,000 kg	22,000 kg
Filled weight of drum	FULL1	13,000 kg	16,000 kg
Emptied weight of drum	EMPTY1	6,000 kg	8,000 kg
Temperature of brine after warm rinse	WBRINET	30 ⁰ C	40 ⁰ C
Warm water rinse temperature	WARMT1	40 ⁰ C	60 ⁰ C
Amount of warm water used	WARMWT	15,000 litres	20,000 litres
Ammonium Chloride added	AMMONIA	120 kg	160 kg
Interval for pH sample	PHTIME	10 mins	20 mins
Second amount of Aquaderm added	AQUAD2	400 kg	600 kg
Second recycle time	RECYCLE2	40 mins	80 mins
Second cold water wash	COLD2	20 mins	30 mins
Second cold water pumping rate	CORATE2	500 litres/mins	700 litres/mins
Second cold water rinse time	COTIME2	18 mins	30 mins
Added salt	SALT	400 kg	700 kg
Mixing time with salt added	SALTIME	8 mins	12 mins
Amount of sulphuric acid added	SULPH	30 litres	40 litres
Amount of Busan added	BUSAN	1.0 litres	3.0 litres
Testing time interval	TESTINT	20 min	30 min
Measured level of salt	MESALT	50	60
Measured level of acid	MEASACID	22 mLitres	26 mLitres

Table 1

Provide a print out of the treatment tables you use. Once the important factors have been identified optimise the process. Enter the settings of the factors for the optimum value of RATIO into columns labelled Optimum and their names into a column labelled 'Factors' into a worksheet named 'Results.mtw'.

Hints

Each time you start Minitab enter the following command at the MTB> prompt in the session window

swap

This will ensure that the macro file %exper will execute correctly. If you have forgotten to do this Minitab will present you with a “* ASSERTION FAILED IN MODULE MRUTIL AT LINE 284200” message. In this situation save anything in the worksheet you want to keep and restart Minitab. Restarting is the only way it will clear itself. Then type the swap command.

For each design

1. Create the treatment table, using the variable names
2. Label a column 'Results'
3. Label a column 'Factors', and list the variables (by name) that you have used in the treatment table.
4. Run the macro

For assignment submission

1. Print outs of the treatment tables used
2. A disk containing the file 'Results.mtw'

File to contain

1. Column labelled 'Factors', listing the names of the variables used.
2. Column labelled 'Optimum', listing the optimum values for each variable

F.2 Macro Files**File: EXPER.MAC**

This macro controls the execution of the process simulation. After the student has entered their ID number a calculation is performed, and from this a set of experimental conditions is selected. Each set of conditions is contained in a separate macro file which is executed from this macro.

GMACRO

```
exper
# macro written to support 43.328 assignment
#
# Written for: D J Barnes
```

```
# Written by: V R Breen
#
# First written : 20/5/97, adopted from 43.340 21/5/97 update
# Last updated : 12/9/97
#
# Run the process simulation
# Note: the path to the macros will need to be altered if they are not in the default
directory
#
# Ask for student ID number to determine which factors are to be used
Note Please your ID number
set c100;
file 'terminal';
nobs = 1.
let k1 = c100(1)
let k2 = k1/100
let k3 = round(k2)
let k3 = k2 - k3
if k3 < 0
let k3 = k3 + 1
endif
let k3 = k3 * 100
erase c100
# Determine which factors the student has selected
%fact
# Determine which set of conditions to use
if k3 < 20
%cos1
elseif k3 < 40
%cos2
elseif k3 < 60
%cos3
elseif k3 < 80
%cos4
else
%cos5
endif
#
# end of macro
ENDMACRO
```

File: FACT.MAC

This macro converts the list of factors used which has been entered into a column called 'Factors' into a numerical form for comparison purposes as Minitab does not permit textual comparisons.

GMACRO

```
fact
# macro written to support 43.328 assignment
```

```
#  
# Written for: D J Barnes  
# Written by: V R Breen  
#  
# First written : 18/5/97, adapted from 43/340 update 20/5/97  
# Last updated : 12/9/97  
#  
# Determine the factors that have been used in this treatment table  
set c200; # Column names as text  
format (a10);  
nobs 38.  
WATER1  
NASO1  
NASO2  
WATERV1  
RATE1  
TEMP1  
TIME1  
RATE2  
WEIGHT1  
AQUAD1  
TETRAP  
DRUM1  
RPM1  
DRUM2  
RPM2  
ALK1  
RECYCLE1  
RATE3  
VOL1  
FULL1  
EMPTY1  
WBRINET  
WARMT1  
WARMWT  
AMMONIA  
PHTIME  
AQUAD2  
RECYCLE2  
COLD2  
CORATE2  
COTIME2  
SALT  
SALTIME  
SULPH  
BUSAN  
TESTINT  
MESALT  
MEASACID  
end
```

```

set c201; # Numeric representation
nobs 38.
1:38
end
convert c200 c201 'Factors' c203
# end of macro
ENDMACRO

```

File : COS1.MAC

This macro controlled the execution of the process simulation under a specific set of conditions.

GMACRO

```

cos1
# macro written to support 43.328 assignment
#
# Written for: D J Barnes
# Written by: V R Breen
#
# Adapted from corresponding file written for 43.340
# First written : 24/7/97
# Last updated : 12/9/97
#
# Determine the coefficients for the factors used in this treatment table
#
let k15 = count (c203) # number of factors
let k16 = count ('StdOrder') # number of runs
#
do k21 = 1:k15
let k22 = 204 + k21
if c203(k21) = 1
let ck22 = 0
elseif c203(k21) = 2
let ck22 = 0
elseif c203(k21) = 3
let ck22 = -511.56 + (4.5*'NASO2') + (-0.0094*'NASO2'**2)
elseif c203(k21) = 4
let ck22 = 0
elseif c203(k21) = 5
let ck22 = 0
elseif c203(k21) = 6
let ck22 = 0
elseif c203(k21) = 7
let ck22 = 0
elseif c203(k21) = 8
let ck22 = 0
elseif c203(k21) = 9
let ck22 = 0
elseif c203(k21) = 10
let ck22 = 0

```

```
elseif c203(k21) = 11
let ck22 = -14.17 * 'TETRAP'
elseif c203(k21) = 12
let ck22 = 1.53 * 'DRUM1'
elseif c203(k21) = 13
let ck22 = -1.8667 * 'RPM1'
elseif c203(k21) = 14
let ck22 = 0.034*'DRUM2'
elseif c203(k21) = 15
let ck22 = 0
elseif c203(k21) = 16
let ck22 = 1.4 * 'ALK1'
elseif c203(k21) = 17
let ck22 = 0
elseif c203(k21) = 18
let ck22 = 0
elseif c203(k21) = 19
let ck22 = 0
elseif c203(k21) = 20
let ck22 = 0.0008 * 'FULL1'
elseif c203(k21) = 21
let ck22 = 0
elseif c203(k21) = 22
let ck22 = 0
elseif c203(k21) = 23
let ck22 = 0
elseif c203(k21) = 24
let ck22 = -0.0007 * 'WARMWT'
elseif c203(k21) = 25
let ck22 = 0.118 * 'AMMONIA'
elseif c203(k21) = 26
let ck22 = 0.315 * 'PHTIME'
elseif c203(k21) = 27
let ck22 = -0.0293 * 'AQUAD2'
elseif c203(k21) = 28
let ck22 = 0
elseif c203(k21) = 29
let ck22 = 0
elseif c203(k21) = 30
let ck22 = 0
elseif c203(k21) = 31
let ck22 = 0
elseif c203(k21) = 32
let ck22 = 0
elseif c203(k21) = 33
let ck22 = -633.3 + (126 * 'SALTIME') + (-6 * 'SALTIME'**2)
elseif c203(k21) = 34
let ck22 = -0.34 * 'SULPH'
elseif c203(k21) = 35
let ck22 = -36.3 + (66 * 'BUSAN') + (-16.5 * 'BUSAN'**2)
```

```

elseif c203(k21) = 36
let ck22 = 0
elseif c203(k21) = 37
let ck22 = 0.3233 * 'MESALT'
elseif c203(k21) = 38
let ck22 = -0.8227 * 'MEASACID'
endif
enddo
#
# calculate results column
# column c204 is the experimental error calculation
#
Random k16 c204; # Experimental error component
Normal 0.0 1.0.
let k24 = 204 + k15
RSum c204-ck24 'Results'
# erase calculation columns
erase c200-ck24
#
let k17 = 0
do k21 = 1:k16
if ('Results'(k21) < 0) or ('Results'(k21) > 100) # Determine final results
let 'Results'(k21) = 0 # all negatives made equal to zero
let k17 = 1
endif # all entries > 100 set to zero
enddo
if k17 = 1
note You have selected factors that have yielded invalid results
note These are represented as zero values in the Results column
endif
# end of macro
ENDMACRO

```

File : COS2.MAC

This macro controlled the execution of the process simulation under a specific set of conditions.

GMACRO

```

cos2
# macro written to support 43.328 assingment
#
# Written for: D J Barnes
# Written by: V R Breen
#
# Adapted form corresponding file from 43.340
# First written : 24/7/97
# Last updated : 2/10/97
#
# Determine the coefficients for the factors used in this treatment table

```

```
#
let k15 = count (c203) # number of factors
let k16 = count ('StdOrder') # number of runs
#
do k21 = 1:k15
let k22 = 204 + k21
if c203(k21) = 1
let ck22 = 0
elseif c203(k21) = 2
let ck22 = 0
elseif c203(k21) = 3
let ck22 = 0
elseif c203(k21) = 4
let ck22 = 0
elseif c203(k21) = 5
let ck22 = 0
elseif c203(k21) = 6
let ck22 = 0
elseif c203(k21) = 7
let ck22 = 0
elseif c203(k21) = 8
let ck22 = 0
elseif c203(k21) = 9
let ck22 = 0.00152 * 'WEIGHT1'
elseif c203(k21) = 10
let ck22 = -33588.8 + (68.6 * 'AQUAD1') + (-0.035 * 'AQUAD1'**2)
elseif c203(k21) = 11
let ck22 = 0
elseif c203(k21) = 12
let ck22 = 0
elseif c203(k21) = 13
let ck22 = 0
elseif c203(k21) = 14
let ck22 = 0.0714 * 'DRUM2'
elseif c203(k21) = 15
let ck22 = -1.1667 * 'RPM2'
elseif c203(k21) = 16
let ck22 = 1.4 * 'ALK1'
elseif c203(k21) = 17
let ck22 = 0
elseif c203(k21) = 18
let ck22 = 0.0214 * 'RATE3'
elseif c203(k21) = 19
let ck22 = -0.0005 * 'VOL1'
elseif c203(k21) = 20
let ck22 = 0
elseif c203(k21) = 21
let ck22 = 0
elseif c203(k21) = 22
let ck22 = 0.38 * 'WBRINET'
```

```

elseif c203(k21) = 23
let ck22 = 0
elseif c203(k21) = 24
let ck22 = 0
elseif c203(k21) = 25
let ck22 = 0
elseif c203(k21) = 26
let ck22 = 0
elseif c203(k21) = 27
let ck22 = -0.0293 * 'AQUAD2'
elseif c203(k21) = 28
let ck22 = 0
elseif c203(k21) = 29
let ck22 = 0
elseif c203(k21) = 30
let ck22 = 0
elseif c203(k21) = 31
let ck22 = 0
elseif c203(k21) = 32
let ck22 = 0
elseif c203(k21) = 33
let ck22 = -633.6 + (126 * 'SALTIME') + (-6 * 'SALTIME'**2)
elseif c203(k21) = 34
let ck22 = -0.34 * 'SULPH'
elseif c203(k21) = 35
let ck22 = -118.8 + (198 * 'BUSAN') + (-66 * 'BUSAN'**2)
elseif c203(k21) = 36
let ck22 = -0.645 * 'TESTINT'
elseif c203(k21) = 37
let ck22 = 0.3233 * 'MESALT'
elseif c203(k21) = 38
let ck22 = -0.8227 * 'MEASACID'
endif
enddo
#
# calculate results column
# column c204 is the experimental error calculation
#
Random k16 c204; # Experimental error component
Normal 0.0 1.0.
let k24 = 204 + k15
RSum c204-ck24 'Results'
# erase calculation columns
erase c200-ck24
#
let k17 = 0
do k21 = 1:k16
if ('Results'(k21) < 0) or ('Results'(k21) > 100) # Determine final results
let 'Results'(k21) = 0 # all negatives made equal to zero
let k17 = 1

```

```

endif                                # all entries > 100 set to zero
enddo
if k17 = 1
note You have selected factors that have yielded invalid results
note These are represented as zero values in the Results column
endif
# end of macro
ENDMACRO

```

File : COS3.MAC

This macro controlled the execution of the process simulation under a specific set of conditions.

```

GMACRO
cos3
# macro written to support 43.328 assignment
#
# Written for: D J Barnes
# Written by: V R Breen
#
# Adapted from corresponding file from 43.340
# First written : 24/7/97
# Last updated : 2/10/97
#
# Determine the coefficients for the factors used in this treatment table
#
let k15 = count (c203) # number of factors
let k16 = count ('StdOrder') # number of runs
#
do k21 = 1:k15
let k22 = 204 + k21
if c203(k21) = 1
let ck22 = 0
elseif c203(k21) = 2
let ck22 = 0.2267 * 'NASO1'
elseif c203(k21) = 3
let ck22 = -511.56 + (4.5 * 'NASO2') + (-0.0094 * 'NASO2'**2)
elseif c203(k21) = 4
let ck22 = 0
elseif c203(k21) = 5
let ck22 = 0
elseif c203(k21) = 6
let ck22 = 0
elseif c203(k21) = 7
let ck22 = 0
elseif c203(k21) = 8
let ck22 = 0
elseif c203(k21) = 9
let ck22 = -0.00154 * 'WEIGHT1'

```

```
elseif c203(k21) = 10
let ck22 = 0
elseif c203(k21) = 11
let ck22 = -2675.2 + (4160 * 'TETRAP') + (-1600 * 'TETRAP'**2)
elseif c203(k21) = 12
let ck22 = 0
elseif c203(k21) = 13
let ck22 = 0
elseif c203(k21) = 14
let ck22 = 0
elseif c203(k21) = 15
let ck22 = 0
elseif c203(k21) = 16
let ck22 = -2.33 * 'ALK1'
elseif c203(k21) = 17
let ck22 = 0
elseif c203(k21) = 18
let ck22 = 0.0214 * 'RATE3'
elseif c203(k21) = 19
let ck22 = -0.0005 * 'VOL1'
elseif c203(k21) = 20
let ck22 = 0.0008 * 'FULL1'
elseif c203(k21) = 21
let ck22 = -0.0021 * 'EMPTY1'
elseif c203(k21) = 22
let ck22 = 0
elseif c203(k21) = 23
let ck22 = 0
elseif c203(k21) = 24
let ck22 = 0
elseif c203(k21) = 25
let ck22 = -679.68 + (10.08 * 'AMMONIA') + (-0.036 * 'AMMONIA'**2)
elseif c203(k21) = 26
let ck22 = -0.63 * 'PHTIME'
elseif c203(k21) = 27
let ck22 = 0.0195 * 'AQUAD2'
elseif c203(k21) = 28
let ck22 = 0
elseif c203(k21) = 29
let ck22 = 0
elseif c203(k21) = 30
let ck22 = 0
elseif c203(k21) = 31
let ck22 = 0
elseif c203(k21) = 32
let ck22 = 0
elseif c203(k21) = 33
let ck22 = 0
elseif c203(k21) = 34
let ck22 = 0
```

```

elseif c203(k21) = 35
let ck22 = 0
elseif c203(k21) = 36
let ck22 = 0.43 * 'TESTINT'
elseif c203(k21) = 37
let ck22 = 0.3233 * 'MESALT'
elseif c203(k21) = 38
let ck22 = -0.8227 * 'MEASACID'
endif
enddo
#
# calculate results column
# column c204 is the experimental error calculation
#
Random k16 c204; # Experimental error component
Normal 0.0 1.0.
let k24 = 204 + k15
RSum c204-ck24 'Results'
# erase calculation columns
erase c200-ck24
#
let k17 = 0
do k21 = 1:k16
if ('Results'(k21) < 0) or ('Results'(k21) > 100) # Determine final results
let 'Results'(k21) = 0 # all negatives made equal to zero
let k17 = 1
endif # all entries > 100 set to zero
enddo
if k17 = 1
note You have selected factors that have yielded invalid results
note These are represented as zero values in the Results column
endif
# end of macro
ENDMACRO

```

File : COS4.MAC

This macro controlled the execution of the process simulation under a specific set of conditions.

GMACRO

```

cos4
# macro written to support 43.328 assignment
#
# Written for: D J Barnes
# Written by: V R Breen
#
# Adapted from corresponding file for 43.340
# First written : 24/7/97
# Last updated : 2/10/97

```

```

#
# Determine the coefficients for the factors used in this treatment table
#
let k15 = count (c203) # number of factors
let k16 = count ('StdOrder') # number of runs
#
do k21 = 1:k15
let k22 = 204 + k21
if c203(k21) = 1
let ck22 = 0
elseif c203(k21) = 2
let ck22 = 0
elseif c203(k21) = 3
let ck22 = -605.06 + (4.875 * 'NASO2') + (-0.0094 * 'NASO2'**2)
elseif c203(k21) = 4
let ck22 = 0
elseif c203(k21) = 5
let ck22 = 0
elseif c203(k21) = 6
let ck22 = 0
elseif c203(k21) = 7
let ck22 = 0
elseif c203(k21) = 8
let ck22 = 0
elseif c203(k21) = 9
let ck22 = 0.00152 * 'WEIGHT1'
elseif c203(k21) = 10
let ck22 = -32906.3 + (67.9 * 'AQUAD1') + (-0.035 * 'AQUAD1'**2)
elseif c203(k21) = 11
let ck22 = -14.17 * 'TETRAP'
elseif c203(k21) = 12
let ck22 = 1.53 * 'DRUM1'
elseif c203(k21) = 13
let ck22 = -1.8667 * 'RPM1'
elseif c203(k21) = 14
let ck22 = 0.0714 * 'DRUM2'
elseif c203(k21) = 15
let ck22 = 0
elseif c203(k21) = 16
let ck22 = 0
elseif c203(k21) = 17
let ck22 = 0
elseif c203(k21) = 18
let ck22 = 0
elseif c203(k21) = 19
let ck22 = 0
elseif c203(k21) = 20
let ck22 = 0
elseif c203(k21) = 21
let ck22 = 0

```

```

elseif c203(k21) = 22
let ck22 = 0.38 * 'WBRINET'
elseif c203(k21) = 23
let ck22 = 0
elseif c203(k21) = 24
let ck22 = 0.0006 * 'WARMWT'
elseif c203(k21) = 25
let ck22 = -0.1575 * 'AMMONIA'
elseif c203(k21) = 26
let ck22 = 0
elseif c203(k21) = 27
let ck22 = 0
elseif c203(k21) = 28
let ck22 = 0
elseif c203(k21) = 29
let ck22 = 0
elseif c203(k21) = 30
let ck22 = 0
elseif c203(k21) = 31
let ck22 = 0
elseif c203(k21) = 32
let ck22 = 0
elseif c203(k21) = 33
let ck22 = 0
elseif c203(k21) = 34
let ck22 = 0.255 * 'SULPH'
elseif c203(k21) = 35
let ck22 = -118.8 + (198 * 'BUSAN') + (-66 * 'BUSAN'**2)
elseif c203(k21) = 36
let ck22 = -0.645 * 'TESTINT'
elseif c203(k21) = 37
let ck22 = 0.3233 * 'MESALT'
elseif c203(k21) = 38
let ck22 = -0.8227 * 'MEASACID'
endif
enddo
#
# calculate results column
# column c204 is the experimental error calculation
#
Random k16 c204; # Experimental error component
Normal 0.0 1.0.
let k24 = 204 + k15
RSum c204-ck24 'Results'
# erase calculation columns
erase c200-ck24
#
let k17 = 0
do k21 = 1:k16
if ('Results'(k21) < 0) or ('Results'(k21) > 100) # Determine final results

```

```

let 'Results'(k21) = 0                # all negatives made equal to zero
let k17 = 1
endif                                # all entries > 100 set to zero
enddo
if k17 = 1
note You have selected factors that have yielded invalid results
note These are represented as zero values in the Results column
endif
# end of macro
ENDMACRO

```

File : COS5.MAC

This macro controlled the execution of the process simulation under a specific set of conditions.

GMACRO

cos5

macro written to support 43.328 assignment

#

Written for: D J Barnes

Written by: V R Breen

#

Adapted from corresponding file from 43.340

First written : 24/7/97

Last updated : 12/9/97

#

Determine the coefficients for the factors used in this treatment table

#

let k15 = count (c203) # number of factors

let k16 = count ('StdOrder') # number of runs

#

do k21 = 1:k15

let k22 = 204 + k21

if c203(k21) = 1

let ck22 = 0

elseif c203(k21) = 2

let ck22 = 0

elseif c203(k21) = 3

let ck22 = 0

elseif c203(k21) = 4

let ck22 = -0.017 * 'WATERV1'

elseif c203(k21) = 5

let ck22 = 0

elseif c203(k21) = 6

let ck22 = 0

elseif c203(k21) = 7

let ck22 = 0

elseif c203(k21) = 8

let ck22 = 0

elseif c203(k21) = 9

```
let ck22 = 0
elseif c203(k21) = 10
let ck22 = 0
elseif c203(k21) = 11
let ck22 = 12.14 * 'TETRAP'
elseif c203(k21) = 12
let ck22 = 0
elseif c203(k21) = 13
let ck22 = 0
elseif c203(k21) = 14
let ck22 = 0.0714 * 'DRUM2'
elseif c203(k21) = 15
let ck22 = -1.1667 * 'RPM2'
elseif c203(k21) = 16
let ck22 = -2.33 * 'ALK1'
elseif c203(k21) = 17
let ck22 = 0
elseif c203(k21) = 18
let ck22 = 0
elseif c203(k21) = 19
let ck22 = 0
elseif c203(k21) = 20
let ck22 = 0.0008 * 'FULL1'
elseif c203(k21) = 21
let ck22 = -0.0021 * 'EMPTY1'
elseif c203(k21) = 22
let ck22 = 0.38 * 'WBRINET'
elseif c203(k21) = 23
let ck22 = 0
elseif c203(k21) = 24
let ck22 = 0
elseif c203(k21) = 25
let ck22 = -1319.68 + (18.56 * 'AMMONIA') + (-0.064 * 'AMMONIA'**2)
elseif c203(k21) = 26
let ck22 = 0
elseif c203(k21) = 27
let ck22 = 0.0195 * 'AQUAD2'
elseif c203(k21) = 28
let ck22 = 0
elseif c203(k21) = 29
let ck22 = 0
elseif c203(k21) = 30
let ck22 = 0
elseif c203(k21) = 31
let ck22 = 0
elseif c203(k21) = 32
let ck22 = 0
elseif c203(k21) = 33
let ck22 = -513.6 + (114 * 'SALTIME') + (-6 * 'SALTIME'**2)
elseif c203(k21) = 34
```

```

let ck22 = -0.34 * 'SULPH'
elseif c203(k21) = 35
let ck22 = -36.3 + (66 * 'BUSAN') + (-16.5 * 'BUSAN'**2)
elseif c203(k21) = 36
let ck22 = 0
elseif c203(k21) = 37
let ck22 = 0.3233 * 'MESALT'
elseif c203(k21) = 38
let ck22 = -0.8227 * 'MEASACID'
endif
enddo
#
# calculate results column
# column c204 is the experimental error calculation
#
Random k16 c204; # Experimental error component
Normal 0.0 1.0.
let k24 = 204 + k15
RSum c204-ck24 'Results'
# erase calculation columns
erase c200-ck24
#
let k17 = 0
do k21 = 1:k16
if ('Results'(k21) < 0) or ('Results'(k21) > 100) # Determine final results
let 'Results'(k21) = 0 # all negatives made equal to zero
let k17 = 1
endif # all entries > 100 set to zero
enddo
if k17 = 1
note You have selected factors that have yielded invalid results
note These are represented as zero values in the Results column
endif
# end of macro
ENDMACRO

```

Assignment marking

File: Mark.mac

Control macro for the marking of assignment submissions. This macro requested the student ID Number and determined which set of experimental conditions (which cos file) were used and therefore which marking schedule to use.

GMACRO

```

mark
# macro written to support 43.328 assignment
#

```

```

# Written for: D J Barnes
# Written by: V R Breen
#
# First written : 8/6/97, adopted from 43.340 25/9/97 update
# Last updated : 25/9/97
#
# Run the process simulation
# Note: the path to the macros will need to be altered if they are not in the default
directory
#
# Ask for student ID number to determine which factors are to be used
Note Please your ID number
set c100;
file 'terminal';
nobs = 1.
let k1 = c100(1)
let k2 = k1/100
let k3 = round(k2)
let k3 = k2 - k3
if k3 < 0
let k3 = k3 + 1
endif
let k3 = k3 * 100
erase c100
# Determine which factors the student has selected
%fact
# Determine which set of conditions to use
if k3 < 20
%mrk1
elseif k3 < 40
%mrk2
elseif k3 < 60
%mrk3
elseif k3 < 80
%mrk4
else
%mrk5
endif
#
# end of macro
ENDMACRO

```

File: mrk1.mac

Mark values taken submitted when cos1.mac was used.

GMACRO

```

mrk1
# macro written to support 43.340 assignment
#

```

```
# Written for: D J Barnes
# Written by: V R Breen
#
# First written : 8/6/97
# Last updated : 8/6/97
#
# Marking of the "Results" worksheet
#
%Factors # converts text to numeric form, stored in c203
let k12 = count ('Factors') # number of factors
#
do k21 = 1:k12
if c203(k21) = 1
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 2
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 3
let c204(k21) = 8
if ('Optimum'(k21) > 237.5) and ('Optimum'(k21) < 262.5)
let c205(k21) = 1
else
let c205(k21) = 0
endif
elseif c203(k21) = 4
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 5
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 6
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 7
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 8
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 9
let c204(k21) = 5
if ('Optimum'(k21) >= 6800) and ('Optimum'(k21) < 7140)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 10
let c204(k21) = 8
if ('Optimum'(k21) > 926.25) and ('Optimum'(k21) < 1023.75)
```

```
let c205(k21) = 1
else
let c205(k21) = 0
endif
elseif c203(k21) = 11
let c204(k21) = 5
if ('Optimum'(k21) > 1.33) and ('Optimum'(k21) <= 1.4)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 12
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 13
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 14
let c204(k21) = 2
if ('Optimum'(k21) > 66.5) and ('Optimum'(k21) <= 70)
let c205(k21) = (1/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 15
let c204(k21) = 2
if ('Optimum'(k21) >= 6) and ('Optimum'(k21) < 6.3)
let c205(k21) = (1/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 16
let c204(k21) = 5
if ('Optimum'(k21) >= 6) and ('Optimum'(k21) < 6.3)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 17
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 18
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 19
let c204(k21) = 2
if ('Optimum'(k21) >= 18000) and ('Optimum'(k21) < 18900)
let c205(k21) = (1/3)
else
let c205(k21) = 0
```

```
endif
elseif c203(k21) = 20
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 21
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 22
let c204(k21) = 5
if ('Optimum'(k21) >= 30) and ('Optimum'(k21) < 31.5)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 23
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 24
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 25
let c204(k21) = 5
if ('Optimum'(k21) >= 120) and ('Optimum'(k21) < 126)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 26
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 27
let c204(k21) = 5
if ('Optimum'(k21) > 570) and ('Optimum'(k21) <= 600)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 28
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 29
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 30
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 31
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 32
```

```
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 33
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 34
let c204(k21) = 5
if ('Optimum'(k21) >= 30) and ('Optimum'(k21) < 31.5)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 35
let c204(k21) = 8
if ('Optimum'(k21) > 1.9) and ('Optimum'(k21) < 2.1)
let c205(k21) = 1
else
let c205(k21) = 0
endif
elseif c203(k21) = 36
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 37
let c204(k21) = 5
if ('Optimum'(k21) > 57) and ('Optimum'(k21) <= 60)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 38
let c204(k21) = 5
if ('Optimum'(k21) >= 22) and ('Optimum'(k21) < 23.1)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
endif
enddo
#
# calculate the marks for factor identification and optimum values
#
let k21 = sum(c204)+sum(c205)
print k21
erase c200-c205
# end of macro
ENDMACRO
```

File: mrk2.mac

Mark values taken submitted when cos2.mac was used.

```
GMACRO
mrk2
# macro written to support 43.340 assignment
#
# Written for: D J Barnes
# Written by: V R Breen
#
# First written : 8/6/97
# Last updated : 8/6/97
#
# Marking of the "Results" worksheet
#
%Factors # converts text to numeric form, stored in c203
let k12 = count ('Factors') # number of factors
#
do k21 = 1:k12
if c203(k21) = 1
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 2
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 3
let c204(k21) = 8
if ('Optimum'(k21) > 237.5) and ('Optimum'(k21) < 262.5)
let c205(k21) = 1
else
let c205(k21) = 0
endif
elseif c203(k21) = 4
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 5
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 6
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 7
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 8
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 9
let c204(k21) = 5
if ('Optimum'(k21) >= 6800) and ('Optimum'(k21) < 7140)
let c205(k21) = (2/3)
else
```

```
let c205(k21) = 0
endif
elseif c203(k21) = 10
let c204(k21) = 8
if ('Optimum'(k21) > 926.25) and ('Optimum'(k21) < 1023.75)
let c205(k21) = 1
else
let c205(k21) = 0
endif
elseif c203(k21) = 11
let c204(k21) = 5
if ('Optimum'(k21) > 1.33) and ('Optimum'(k21) <= 1.4)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 12
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 13
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 14
let c204(k21) = 2
if ('Optimum'(k21) > 66.5) and ('Optimum'(k21) <= 70)
let c205(k21) = (1/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 15
let c204(k21) = 2
if ('Optimum'(k21) >= 6) and ('Optimum'(k21) < 6.3)
let c205(k21) = (1/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 16
let c204(k21) = 5
if ('Optimum'(k21) >= 6) and ('Optimum'(k21) < 6.3)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 17
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 18
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 19
```

```
let c204(k21) = 2
if ('Optimum'(k21) >= 18000) and ('Optimum'(k21) < 18900)
let c205(k21) = (1/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 20
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 21
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 22
let c204(k21) = 5
if ('Optimum'(k21) >= 30) and ('Optimum'(k21) < 31.5)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 23
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 24
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 25
let c204(k21) = 5
if ('Optimum'(k21) >= 120) and ('Optimum'(k21) < 126)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 26
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 27
let c204(k21) = 5
if ('Optimum'(k21) > 570) and ('Optimum'(k21) <= 600)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 28
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 29
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 30
let c204(k21) = 0
```

```

let c205(k21) = 0
elseif c203(k21) = 31
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 32
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 33
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 34
let c204(k21) = 5
if ('Optimum'(k21) >= 30) and ('Optimum'(k21) < 31.5)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 35
let c204(k21) = 8
if ('Optimum'(k21) > 1.9) and ('Optimum'(k21) < 2.1)
let c205(k21) = 1
else
let c205(k21) = 0
endif
elseif c203(k21) = 36
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 37
let c204(k21) = 5
if ('Optimum'(k21) > 57) and ('Optimum'(k21) <= 60)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 38
let c204(k21) = 5
if ('Optimum'(k21) >= 22) and ('Optimum'(k21) < 23.1)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
endif
enddo
#
# calculate the marks for factor identification and optimum values
#
let k21 = sum(c204)+sum(c205)
print k21
erase c200-c205
# end of macro

```

ENDMACRO

File: mrk3.mac

Mark values taken submitted when cos3.mac was used.

GMACRO

mrk3

macro written to support 43.340 assignment

#

Written for: D J Barnes

Written by: V R Breen

#

First written : 8/6/97

Last updated : 8/6/97

#

Marking of the "Results" worksheet

#

%Factors # converts text to numeric form, stored in c203

let k12 = count ('Factors') # number of factors

#

do k21 = 1:k12

if c203(k21) = 1

let c204(k21) = 0

let c205(k21) = 0

elseif c203(k21) = 2

let c204(k21) = 0

let c205(k21) = 0

elseif c203(k21) = 3

let c204(k21) = 8

if ('Optimum'(k21) > 237.5) and ('Optimum'(k21) < 262.5)

let c205(k21) = 1

else

let c205(k21) = 0

endif

elseif c203(k21) = 4

let c204(k21) = 0

let c205(k21) = 0

elseif c203(k21) = 5

let c204(k21) = 0

let c205(k21) = 0

elseif c203(k21) = 6

let c204(k21) = 0

let c205(k21) = 0

elseif c203(k21) = 7

let c204(k21) = 0

let c205(k21) = 0

elseif c203(k21) = 8

let c204(k21) = 0

let c205(k21) = 0

```
elseif c203(k21) = 9
let c204(k21) = 5
if ('Optimum'(k21) >= 6800) and ('Optimum'(k21) < 7140)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 10
let c204(k21) = 8
if ('Optimum'(k21) > 926.25) and ('Optimum'(k21) < 1023.75)
let c205(k21) = 1
else
let c205(k21) = 0
endif
elseif c203(k21) = 11
let c204(k21) = 5
if ('Optimum'(k21) > 1.33) and ('Optimum'(k21) <= 1.4)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 12
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 13
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 14
let c204(k21) = 2
if ('Optimum'(k21) > 66.5) and ('Optimum'(k21) <= 70)
let c205(k21) = (1/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 15
let c204(k21) = 2
if ('Optimum'(k21) >= 6) and ('Optimum'(k21) < 6.3)
let c205(k21) = (1/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 16
let c204(k21) = 5
if ('Optimum'(k21) >= 6) and ('Optimum'(k21) < 6.3)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 17
let c204(k21) = 0
```

```
let c205(k21) = 0
elseif c203(k21) = 18
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 19
let c204(k21) = 2
if ('Optimum'(k21) >= 18000) and ('Optimum'(k21) < 18900)
let c205(k21) = (1/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 20
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 21
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 22
let c204(k21) = 5
if ('Optimum'(k21) >= 30) and ('Optimum'(k21) < 31.5)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 23
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 24
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 25
let c204(k21) = 5
if ('Optimum'(k21) >= 120) and ('Optimum'(k21) < 126)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 26
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 27
let c204(k21) = 5
if ('Optimum'(k21) > 570) and ('Optimum'(k21) <= 600)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 28
let c204(k21) = 0
let c205(k21) = 0
```

```
elseif c203(k21) = 29
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 30
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 31
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 32
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 33
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 34
let c204(k21) = 5
if ('Optimum'(k21) >= 30) and ('Optimum'(k21) < 31.5)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 35
let c204(k21) = 8
if ('Optimum'(k21) > 1.9) and ('Optimum'(k21) < 2.1)
let c205(k21) = 1
else
let c205(k21) = 0
endif
elseif c203(k21) = 36
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 37
let c204(k21) = 5
if ('Optimum'(k21) > 57) and ('Optimum'(k21) <= 60)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 38
let c204(k21) = 5
if ('Optimum'(k21) >= 22) and ('Optimum'(k21) < 23.1)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
endif
enddo
#
# calculate the marks for factor identification and optimum values
```

```
#
let k21 = sum(c204)+sum(c205)
print k21
erase c200-c205
# end of macro
ENDMACRO
```

File mrk4.mac

Mark values taken submitted when cos4.mac was used.

```
GMACRO
mrk4
# macro written to support 43.340 assignment
#
# Written for: D J Barnes
# Written by: V R Breen
#
# First written : 8/6/97
# Last updated : 8/6/97
#
# Marking of the "Results" worksheet
#
%Factors # converts text to numeric form, stored in c203
let k12 = count ('Factors') # number of factors
#
do k21 = 1:k12
if c203(k21) = 1
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 2
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 3
let c204(k21) = 8
if ('Optimum'(k21) > 237.5) and ('Optimum'(k21) < 262.5)
let c205(k21) = 1
else
let c205(k21) = 0
endif
elseif c203(k21) = 4
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 5
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 6
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 7
```

```
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 8
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 9
let c204(k21) = 5
if ('Optimum'(k21) >= 6800) and ('Optimum'(k21) < 7140)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 10
let c204(k21) = 8
if ('Optimum'(k21) > 926.25) and ('Optimum'(k21) < 1023.75)
let c205(k21) = 1
else
let c205(k21) = 0
endif
elseif c203(k21) = 11
let c204(k21) = 5
if ('Optimum'(k21) > 1.33) and ('Optimum'(k21) <= 1.4)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 12
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 13
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 14
let c204(k21) = 2
if ('Optimum'(k21) > 66.5) and ('Optimum'(k21) <= 70)
let c205(k21) = (1/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 15
let c204(k21) = 2
if ('Optimum'(k21) >= 6) and ('Optimum'(k21) < 6.3)
let c205(k21) = (1/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 16
let c204(k21) = 5
if ('Optimum'(k21) >= 6) and ('Optimum'(k21) < 6.3)
let c205(k21) = (2/3)
```

```
else
let c205(k21) = 0
endif
elseif c203(k21) = 17
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 18
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 19
let c204(k21) = 2
if ('Optimum'(k21) >= 18000) and ('Optimum'(k21) < 18900)
let c205(k21) = (1/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 20
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 21
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 22
let c204(k21) = 5
if ('Optimum'(k21) >= 30) and ('Optimum'(k21) < 31.5)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 23
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 24
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 25
let c204(k21) = 5
if ('Optimum'(k21) >= 120) and ('Optimum'(k21) < 126)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 26
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 27
let c204(k21) = 5
if ('Optimum'(k21) > 570) and ('Optimum'(k21) <= 600)
let c205(k21) = (2/3)
else
```

```
let c205(k21) = 0
endif
elseif c203(k21) = 28
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 29
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 30
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 31
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 32
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 33
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 34
let c204(k21) = 5
if ('Optimum'(k21) >= 30) and ('Optimum'(k21) < 31.5)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 35
let c204(k21) = 8
if ('Optimum'(k21) > 1.9) and ('Optimum'(k21) < 2.1)
let c205(k21) = 1
else
let c205(k21) = 0
endif
elseif c203(k21) = 36
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 37
let c204(k21) = 5
if ('Optimum'(k21) > 57) and ('Optimum'(k21) <= 60)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 38
let c204(k21) = 5
if ('Optimum'(k21) >= 22) and ('Optimum'(k21) < 23.1)
let c205(k21) = (2/3)
else
let c205(k21) = 0
```

```
endif
endif
enddo
#
# calculate the marks for factor identification and optimum values
#
let k21 = sum(c204)+sum(c205)
print k21
erase c200-c205
# end of macro
ENDMACRO
```

File:mrk5.mac

Mark values taken submitted when cos5.mac was used.

```
GMACRO
mrk5
# macro written to support 43.340 assignment
#
# Written for: D J Barnes
# Written by: V R Breen
#
# First written : 8/6/97
# Last updated : 8/6/97
#
# Marking of the "Results" worksheet
#
%Factors # converts text to numeric form, stored in c203
let k12 = count ('Factors') # number of factors
#
do k21 = 1:k12
if c203(k21) = 1
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 2
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 3
let c204(k21) = 8
if ('Optimum'(k21) > 237.5) and ('Optimum'(k21) < 262.5)
let c205(k21) = 1
else
let c205(k21) = 0
endif
elseif c203(k21) = 4
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 5
let c204(k21) = 0
```

```
let c205(k21) = 0
elseif c203(k21) = 6
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 7
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 8
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 9
let c204(k21) = 5
if ('Optimum'(k21) >= 6800) and ('Optimum'(k21) < 7140)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 10
let c204(k21) = 8
if ('Optimum'(k21) > 926.25) and ('Optimum'(k21) < 1023.75)
let c205(k21) = 1
else
let c205(k21) = 0
endif
elseif c203(k21) = 11
let c204(k21) = 5
if ('Optimum'(k21) > 1.33) and ('Optimum'(k21) <= 1.4)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 12
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 13
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 14
let c204(k21) = 2
if ('Optimum'(k21) > 66.5) and ('Optimum'(k21) <= 70)
let c205(k21) = (1/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 15
let c204(k21) = 2
if ('Optimum'(k21) >= 6) and ('Optimum'(k21) < 6.3)
let c205(k21) = (1/3)
else
let c205(k21) = 0
```

```
endif
elseif c203(k21) = 16
let c204(k21) = 5
if ('Optimum'(k21) >= 6) and ('Optimum'(k21) < 6.3)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 17
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 18
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 19
let c204(k21) = 2
if ('Optimum'(k21) >= 18000) and ('Optimum'(k21) < 18900)
let c205(k21) = (1/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 20
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 21
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 22
let c204(k21) = 5
if ('Optimum'(k21) >= 30) and ('Optimum'(k21) < 31.5)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 23
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 24
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 25
let c204(k21) = 5
if ('Optimum'(k21) >= 120) and ('Optimum'(k21) < 126)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 26
let c204(k21) = 0
let c205(K21) = 0
```

```
elseif c203(k21) = 27
let c204(k21) = 5
if ('Optimum'(k21) > 570) and ('Optimum'(k21) <= 600)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 28
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 29
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 30
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 31
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 32
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 33
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 34
let c204(k21) = 5
if ('Optimum'(k21) >= 30) and ('Optimum'(k21) < 31.5)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 35
let c204(k21) = 8
if ('Optimum'(k21) > 1.9) and ('Optimum'(k21) < 2.1)
let c205(k21) = 1
else
let c205(k21) = 0
endif
elseif c203(k21) = 36
let c204(k21) = 0
let c205(k21) = 0
elseif c203(k21) = 37
let c204(k21) = 5
if ('Optimum'(k21) > 57) and ('Optimum'(k21) <= 60)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
elseif c203(k21) = 38
```

```
let c204(k21) = 5
if ('Optimum'(k21) >= 22) and ('Optimum'(k21) < 23.1)
let c205(k21) = (2/3)
else
let c205(k21) = 0
endif
endif
enddo
#
# calculate the marks for factor identification and optimum values
#
let k21 = sum(c204)+sum(c205)
print k21
erase c200-c205
# end of macro
ENDMACRO
```