

```
/*
 * canonize.c
 *
 * This file provides the function
 *
 *      FuncResult canonize(Triangulation *manifold);
 *
 * canonize() is a shell, which calls the functions
 *
 *      proto_canonize()          [found in canonize_part_1.c]
 *      canonical_retriangulation() [found in canonize_part_2.c]
 *
 * The purpose of these functions is explained in the code below.
 * For the mathematical details, please see canonize_part_1.c
 * and canonize_part_2.c.
 *
 * canonize() does not preserve the original Triangulation;
 * if you need to keep it, make a copy with copy_triangulation()
 * before calling canonize().
 */

#include "kernel.h"

FuncResult canonize(
    Triangulation *manifold)
{
    /*
     * Apply the tilt theorem to compute a Triangulation
     * which is a subdivision of the canonical cell decomposition.
     * Please see canonize_part_1.c for details.
     */

    if (proto_canonize(manifold) == func_failed)
        return func_failed;

    /*
     * Replace the given subdivision of the canonical cell
     * decomposition with the canonical retriangulation.
     * This operation introduces finite vertices whenever
     * the canonical cell decomposition is not a triangulation
     * to begin with. Please see canonize_part_2.c for details.
     */

    canonical_retriangulation(manifold);

    return func_OK;
}
```