# Systems Programming: Practical 3 — Input and Output

### A Background reading on Input and Output:

Familiarize yourself with file I/O and the fopen() and fclose() functions in Chapter 12 (particularly 12.1 and 12.4) from http://www.eskimo.com/~scs/cclass/notes/top.html and Chapter 17 (particularly 17.1) on formatted text input from files using fscanf() and output with fprintf() from http://www.eskimo.com/~scs/cclass/int/sx3a.html.

### B Simple file reading and writing

Test your understanding of file reading by producing a C program to read in the provided file read.txt. This file contains three lines: name, age and height. Your C program should read these three values into appropriate variable types.

Write a new program that outputs your own name, age and height to a file.

# C File I/O: reading text files, data structures, calculations and writing text files

The file input.txt supplied for the practical contains data relating to five three-dimensional triangles. The data provided on each line are the three corners as 3D points in anti-clockwise order. Read in each triangle and calculate its area<sup>1</sup>. Use a struct type<sup>2</sup> to store the triangle data and the calculated results. Write out a new file called output.txt that has the triangles as above, but with the addition of the value of the area.

Hint: each triangle has a constant z value. Therefore you can ignore the z coordinate when you calculate the area and are able to treat the triangles as 2D shapes. It's not that hard to include the z parameter if you want.

## D Background Reading, Command line parameters

Look at the section on command-line parameters to main methods: http://www.eskimo.com/~scs/cclass/notes/sx13.html

## E Adding command line parameters to your program

Adapt your program (from C) to use command line parameters. The first is the input file name and the second is the output filename. If only one command-line parameter is provided, use it as the input file name and derive the output file name by adding ".out" to the end of it. If no command line parameters are provided, or the files are not readable/writable, prompt the user for file names from input with scanf(). Offer a default option of input.txt and output.txt to be used if the user just hits return when prompted. Get this all to work on Linux.

#### F Extension Exercise

If all that is too easy for you, then try to make your program a bit more professional by specifying the input and output files with command line options -f (for input file) and -o (for output file). Use the getopt() function from the standard C library to parse the command line so that you can specify the options in any order. You can find the documentation for getopt() as part of the documentation for the standard C library (libc): http://www.gnu.org/software/libc/manual/html\_node/Getopt.html#Getopt. Alternatively you could use argp: http://www.gnu.org/software/libc/manual/html\_node/Argp.html. (N.B. getopt() and argp are not part of the C standard.)

<sup>&</sup>lt;sup>1</sup>The area of a triangle can be calculated using one of the formulas available here: http://www.mathopenref.com/coordtrianglearea.html.

<sup>&</sup>lt;sup>2</sup>http://www.lysator.liu.se/c/bwk-tutor.html#structure