EE 5301 Fall 2017 Mini Project #1

Deadlines

Part 1: Nov 10, 11:59pm

Part 2: Nov 20, 11:59pm

(The deadlines are far away, but you need to start today!)

Description of the mini-project

In this project, you will build a slicing floorplan for a set of blocks using a simulated annealing engine. You will need to

(1) Read in and parse the input description

(2) Build and tune the parameters of a simulated annealing engine

a. Represent the floorplan slicing tree the form of a string

b. Evaluate the cost of the slicing tree [an area cost is good enough]

c. Define a set of moves, ensuring that each move results in a legal string and the set of moves is capable of reaching the entire solution space

d. Determine an initial and final temperature, the cooling schedule, and the number of moves per temperature

Part 1) Use the hard block benchmark files to build a slicing floorplan with hard blocks
Part 2) Do the same for the soft block benchmark files

Input format:

- For the hard macro benchmarks, for each block, you are given the coordinates of the four corners of the block when the bottom left corner is at (0,0). This defines the block size.
- For the soft macro benchmarks, each block name is succeeded by the area, the minimum aspect ratio, and the maximum aspect ratio (where aspect ratio is the ratio of width to height). In principle, you could have an infinite set of combinations of width x height = area, but please consider just three: minimum aspect ratio, maximum aspect ratio, and aspect ratio = 1. Of these, you can drop the last if an aspect ratio of 1 is not allowed. And obviously if the minimum and maximum aspect ratio are identical, then you have only one valid option.

All benchmark files are provided on Moodle

Output format:

Please also print out the total area of the floor plan in the first line of the output file and the amount of empty area (not occupied by blocks) in the second line of the file.

The floorplanning result corresponding to the input file "benchmark_name.blocks" should be printed into the file "benchmark_name.out." Each block should be listed on a separate line. The name of the block should be followed by the x and y coordinates of the bottom left (BL) and the upper right (UR) corner of each block, i.e.,

blockname (BL_x_coordinate, BL_y_coordinate) (UR_x_coordinate, UR_y_coordinate)

Please follow a convention of keeping the bottom left corner of the entire floor plan at the origin, (0,0).

Specifics
Ideally, you should write these programs using C++ (but C will be acceptable too). Your task may be eased by using the C++ Standard Template Library (STL): using STL is not required, but is highly encouraged. A simple outline of the STL is provided on the Moodle page, but you will find many online resources on this topic (e.g., SGI's Standard Template Library – click on the "Index" link. "Table of contents" is very useful too).

You may develop your code on the platform of your choice **BUT** your program should compile on a Unix/Linux platform (with gcc or g++).

Your solution should be capable of floorplan with 500 blocks, and your data structures should be set up accordingly.

Submission Process
Please submit your code via Moodle as one .zip/.rar/.gz/.tgz file containing all source and header files. Your archive should include

- A file called README, listing a brief (one-sentence), logically-organized description of each files within the archive.
- A Makefile to compile your code into an executable: typing "make" should produce an executable called "parser". If you search, you will find plenty of resources online for building a makefile. Please do not submit the compiled executable file.