Homework Assignment #5:
Programming Exercise #1

Due Date & Time

2:10PM Tuesday, April 17, 2018. Late submission will be penalized by 20% for each working day overdue.

Problem Description

Implement an algorithm that computes the skyline of a list of buildings, where the representations of a building and a skyline are as discussed in class. The height and width of a building can be real numbers; your implementation should allow this possibility.

You may assume that there are at most 1000 buildings in each input. The first line of an input is an integer n, indicating the number of buildings. It is followed by n triples of real numbers, each triple in a separate line, indicating the coordinates of a building.

Sample input:

```
8
(1,11,5)
(2,6,7)
(3,13,9)
(12,7,16)
(14,3,25)
(19,18,22)
(23,13,29)
(24,4,28)
```

Correct output for the sample input:

```
(1,11,3,13,9,0,12,7,16,3,19,18,22,3,23,13,29)
```

Notes

This assignment constitutes 4% of your grade. You may discuss the problem with others, but copying code is strictly forbidden. **Some of you may be requested to demonstrate your program.**

Submission Guidelines

- Pack everything, excluding compiler-generated files, in a .zip file, named with the pattern "b057050xx-alg2018-hw5.zip".

- Upload the .zip file to the Ceiba course site for Algorithms 2018:

  `https://ceiba.ntu.edu.tw/1062alg2018`.

- If you use a Makefile, make sure that it outputs "`hw5`". Otherwise, make sure that the whole application can be compiled by a single command like "`gcc hw5.c`", "`g++ hw5.cpp`", or "`javac hw5.java`".

Grading

Your work will be graded according to its correctness and presentation. Before submission, you should have tested your program on several input cases. You should organize and document your program in such a way that other programmers, for example your classmates, can understand it. In the documentation of your program, you should describe how you have applied the algorithmic techniques, in particular design by induction, learned in class. For example, if you choose to use the merging of two skylines as a building block, try to elaborate on how induction has helped in the design of the merging procedure.

Below is a more specific grading policy:

Criteria	Score
incomplete or doesn't compile	≤ 20
complete, compiles, but with major errors	≤ 40
correct but with an $O(n^2)$-time algorithm	≤ 80
correct and with an $O(n \log n)$-time algorithm	≤ 100
explanation of algorithmic techniques applied	$+ \leq 10$