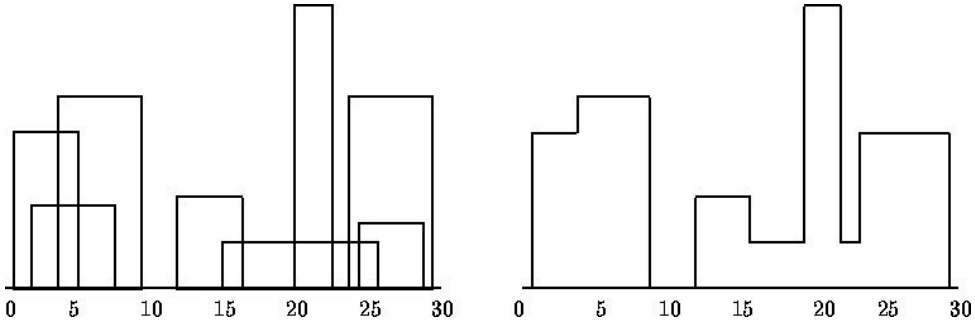
CENG 315

Programming Homework 1
The Skyline Problem

Due: Oct 22th, 2010 23:55

You are to design a program to assist an architect in drawing the skyline of a city given the locations of the buildings in the city. To make the problem tractable, all buildings are rectangular in shape and they share a common bottom (the city they are built in is very flat). The city is also viewed as two-dimensional. A building is specified by an ordered triple (L_i, H_i, R_i) where L_i and R_i are left and right coordinates, respectively, of building i and H_i is the height of the building.

In the diagram below buildings are shown on the left with triples *(1,11,5), (2,6,7), (3,13,9), (12,7,16), (14,3,25), (19,18,22), (23,13,29), (24,4,28)*. Its skyline, shown on the right, is represented by the sequence: *(1, **11**, 3, **13**, 9, **0**, 12, 7, 16, **3**, 19, **18**, 22, **3**, 23, **13**, 29, **0**)*



Input Specification

You will read your input from standard input.

First line of input will contain an integer N, specifying number of buildings. The following N lines will contain three integers, L_i, H_i, R_i (left coordinate, height, right coordinate of building i).

The input does not have to be sorted.

Output Specification

You will write your output to standard output.

First line will contain number of critical points in the skyline (M). The following M lines will contain two integers each: coordinate and height of that critical point. Critical points must be sorted by their coordinates.

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
```

Sample Output

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

Limits

1 <= N <= 100,000
0 < L_i < R_i < 2,000,000,000
0 < H_i < 2,000,000,000
Time limit = **1 second**

Submission / Grading

Submit your *hw1.c* / *hw1.cpp* files through COW. An autograder is available which will grade your homework with real inputs – the grade you get from autograder is final (see the exception below). You can use the autograder up to 3 times per day (so start early!).

Notice that there is a time limit – think about your algorithm's complexity before you code.

Deadline is Oct 22th, 23:55. Late submission is not allowed.

All work should be done individually. Also we know that there are some solutions to this problem on the internet and we are aware of them. We will check your homeworks for cheating/plagiarism after the due date – in case you cheat, you will get 0 from all your homeworks and disciplinary action will be taken.