Computer Science II Program #4: Skyline
Input: Standard in, Output: Standard out

Skyline Problem

The following problem is taken from the following web page: http://acm.uva.es/p/v1/105.html. It is from an archive of over 1000 programming contest problems. I have made slight changes to the description for the purposes of this assignment. **The most important of these changes is that I am requiring you to implement the algorithm shown in class to solve this problem.** The rest of the changes are inserted into the edited problem statement below. As always, include a detailed header comment and other ample comments in your code and observe good programming practices.

The Problem

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) the 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).
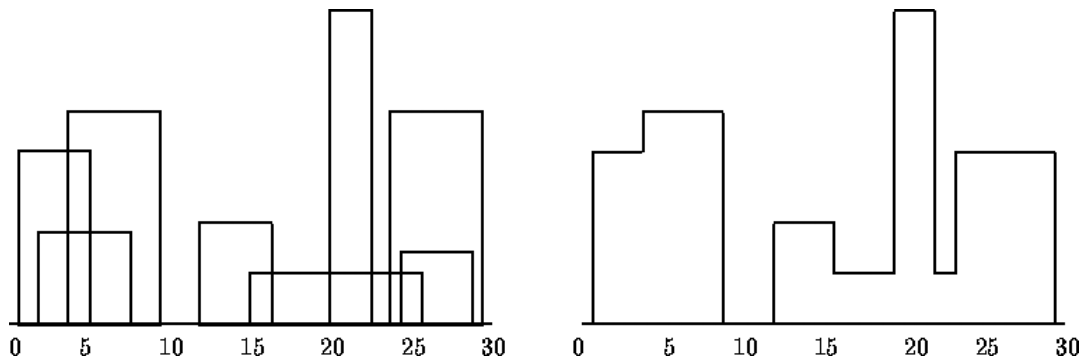
Input Format

The first line of each input case will contain a positive integer n, the number of buildings for that input case. The following n lines of that input case will contain information about each building, one building per line. Each line contains three integers separated by spaces, the left coordinate, height, and right coordinate of that building, respectively. All left and right coordinates of buildings are positive integers less than or equal to 1,000,000, all height coordinates will be positive integers less than or equal to 100,000 and there will be at least one and at most 10,000 buildings in the input file. Each building triple is on a line by itself in the input file. All integers in a triple are separated by one or more spaces. The triples will be sorted by L_i, the left x-coordinate of the building, so the building with the smallest left x-coordinate is first in the input file. A value of n=0 will signify the end of the input.

The Output

Please output to the screen. The output to each case should be on a line by itself. Thus, the number of lines in the output should be the same as the number of cases in the input file. The output for a single case should consist of the vector that describes the skyline as shown in the example above. In the skyline vector $(v_1, v_2, v_3, \ldots, v_n)$ the v_i such that i is an even number represent a horizontal line (height). The v_i such that i is an odd number represent a vertical line (x-coordinate). The skyline vector should represent the ``path'' taken, for example, by a bug starting at the minimum x-coordinate and traveling horizontally and vertically over all the lines that define the skyline. Thus the last entry in the skyline vector will be a 0. The coordinates must be separated by a blank space.

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
2
1  11  5
8  10  9
0
```

Sample Output

```
1 11 3 13 9 0 12 7 16 3 19 18 22 3 23 13 29 0
1 11 5 0 8 10 9 0
```

Deliverables

You must turn in your solution to the problem, **skyline.java**, over WebCourses2/Canvas.