

Overview: Construct a treap

Description: A treap is a hybrid data structure - it is both a binary search tree and a heap. More specifically, each element has two attributes: a value and a priority. A treap is a binary search tree with respect to the values of its elements. In other words, an element's value is greater than that of every element in its left subtree and less than that of every element in its right subtree. It is also a min-heap with respect to the elements' priorities; an element's priority must be greater than that of its parent. Given a set of n elements, there is a unique treap that stores these elements. Construct this treap and print out its pre-order traversal. At each node, first print the node's value then priority, then print out its left subtree and then its right subtree. In order to expose the treap's characteristics, there will be at least 5 elements.

Filename: bug5.{java, cpp, py}

Input: The first line contains a single integer n , the number of elements. The next n lines describe each element. Each line will contain two space separated integers, the element's value and priority, respectively.

Output: Print n lines representing the pre-order traversal of the treap. Each element should be printed on its own line and in the same format as the input.

Assumptions: All input will be valid.
 $5 \leq n \leq 1000$
The priority and value will be between 1 and 1,000,000 inclusive
Every element's priority and value will be unique

Sample Input: 5
5 1
9 4
2 2
1 5
6 3

Sample Output: 5 1
2 2
1 5
6 3
9 4