Guarantrees

locked

by volmahesh

Construct a Data Structure with the following operations:

1. `insert(k)` - Inserts an element into the DS.
2. `delete(k)` - Deletes an element from the DS, if it exists.
3. `min(k)` - Finds the k-th minimum element from the DS. Returns that element's value.

All of the above operations must **guarantee** a worst case complexity of $O(\log n)$

We expect you to get familiar with C++ classes and use them in this assignment.
At the same time, try to write well-commented code. We will appreciate it when grading your submission.

Input Format

```
N (number of commands/lines)
<i|d|m> num ( i=insert ; d=delete ; m=min )
...
...
...
```

Constraints

$N \leq 10^6$

For i,d - `num` is a valid 32-bit integer.

For m - `num` is a valid rank i.e., there are atleast `num` elements in the DS at that time.

Output Format

```
< integer from m command >
...
...
...
```

Sample Input

```
12
i 10
i 21
i 13
i 7
i 5
i 8
i 1020
m 5
d 10
m 5
d 21
m 5
```

Sample Output

```
13
21
1020
```

Explanation

After 7 inserts, the sorted order of elements is `5,7,8,10,13,21,1020`.
The 5th min element is 13, which is printed.
10 is deleted now. This makes the list `5,7,8,13,21,1020`.
The 5th min element is now 21, which is printed.
21 is deleted now. This makes the list `5,7,8,13,1020`.
The 5th min element is finally 1020, which is printed.

Current Buffer (saved locally, editable) C++

```
#include <cmath>
#include <cstdio>
#include <vector>
#include <iostream>
#include <algorithm>
using namespace std;


int main() {
    /* Enter your code here. Read input from STDIN. Print output to STDOUT */
    return 0;
}

```

Line: 1 Col: 1

Upload Code as File Test against custom input Run Code Submit Code