Sets-STL ☆

20/44 challenges solved

Rank: **17617** | Points: **301.25**

Problem Submissions Leaderboard

Sets are a part of the C++ STL. Sets are containers that store unique elements following a specific order. Here are some of the frequently used member functions of sets:

- Declaration:

```
set<int>s; //Creates a set of integers.
```

- Size:

```
int length=s.size(); //Gives the size of the set.
```

- Insert:

```
s.insert(x); //Inserts an integer x into the set s.
```

- Erasing an element:

```
s.erase(val); //Erases an integer val from the set s.
```

- Finding an element:

```
set<int>::iterator itr=s.find(val); //Gives the iterator to the element val if it is found otherwise r
Ex: set<int>::iterator itr=s.find(100); //If 100 is not present then it==s.end().
```

To know more about sets click Here. Coming to the problem, you will be given Q queries. Each query is of one of the following three types:

$1\ x$: Add an element x to the set.
$2\ x$: Delete an element x from the set. (If the number x is not present in the set, then do nothing).
$3\ x$: If the number x is present in the set, then print "Yes"(without quotes) else print "No"(without quotes).

Input Format

The first line of the input contains Q where Q is the number of queries. The next Q lines contain 1 query each. Each query consists of two integers y and x where y is the type of the query and x is an integer.

Constraints

$1 <= Q <= 10^5$
$1 <= y <= 3$
$1 <= x <= 10^9$

Output Format

For queries of type 3 print "Yes"(without quotes) if the number x is present in the set and if the number is not present, then print

"No"(without quotes).

Each query of type **3** should be printed in a new line.

Sample Input

```
8
1 9
1 6
1 10
1 4
3 6
3 14
2 6
3 6
```

Sample Output

```
Yes
No
No
```

C++

```
13      cin >> n;
14
15      for(int i=0;i<n;i++)
16      {
17          cin >> a >> b;
18          switch(a)
19          {
20              case 1:
21                s.insert(b);
22                break;
23
24              case 2:
25                  s.erase(b);
26                  break;
27
28              case 3:
29                //set<int>::iterator itr2 = s.find(b);
30                  auto itr = s.find(b);
31                  if(itr != s.end())
32                      cout << "Yes"<<endl;
33                  else
34                      cout << "No" << endl;
35                break;
36          };
37      }
38      return 0;
39  }
40
41
42
43
```

Line: 34 Col: 32

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

Status: Accepted

Congratulations

You solved this challenge. Would you like to challenge your friends?

Test case 0
Test case 1
Test case 2
Test case 3
Test case 4
Test case 5
Test case 6
Test case 7
Test case 8

Hidden Test Case

Unlock this testcase for 5 hackos.

Unlock