

LeetCode

Explore

Problems

Interview

Contest

Discuss

Store

LeetCode is hiring! Apply NOW.

5

Description

Solution

Discuss (458)

Submissions

i

Java

Autocomplete

i

{ }

↺

⌛

⌕

Easy

315

2402

Add to List

Share

There is a stream of n ($idKey$, $value$) pairs arriving in an **arbitrary** order, where $idKey$ is an integer between 1 and n and $value$ is a string. No two pairs have the same id .

Design a stream that returns the values in **increasing order of their IDs** by returning a **chunk** (list) of values after each insertion. The concatenation of all the **chunks** should result in a list of the sorted values.

Implement the `OrderedStream` class:

- `OrderedStream(int n)` Constructs the stream to take n values.
- `String[] insert(int idKey, String value)` Inserts the pair ($idKey$, $value$) into the stream, then returns the **largest possible chunk** of currently inserted values that appear next in the order.

Example:

Input

```
["OrderedStream", "insert", "insert", "insert", "insert", "insert"]  
[[5], [3, "ccccc"], [1, "aaaaa"], [2, "bbbbbb"], [5, "eeeeee"], [4, "dddddd"]]
```

Output

```
[null, [], ["aaaaa"], ["bbbbbb", "ccccc"], [], ["dddddd", "eeeeee"]]
```

Explanation

```
// Note that the values ordered by ID is ["aaaaa", "bbbbbb", "ccccc", "dddddd", "eeeeee"].  
OrderedStream os = new OrderedStream(5);  
os.insert(3, "ccccc"); // Inserts (3, "ccccc"), returns [].  
os.insert(1, "aaaaa"); // Inserts (1, "aaaaa"), returns ["aaaaa"].  
os.insert(2, "bbbbbb"); // Inserts (2, "bbbbbb"), returns ["bbbbbb", "ccccc"].  
os.insert(5, "eeeeee"); // Inserts (5, "eeeeee"), returns [].  
os.insert(4, "dddddd"); // Inserts (4, "dddddd"), returns ["dddddd", "eeeeee"].  
// Concatenating all the chunks returned:  
// [] + ["aaaaa"] + ["bbbbbb", "ccccc"] + [] + ["dddddd", "eeeeee"] = ["aaaaa", "bbbbbb", "ccccc", "dddddd", "eeeeee"]  
// The resulting order is the same as the order above.
```

Constraints:

- $1 \leq n \leq 1000$
- $1 \leq id \leq n$
- $value.length == 5$
- $value$ consists only of lowercase letters.
- Each call to `insert` will have a unique id .
- Exactly n calls will be made to `insert`.

Accepted 54,141 Submissions 63,422

Seen this question in a real interview before?

Yes

No

Companies

i

Related Topics

Similar Questions

Show Hint 1

Show Hint 2

Show Hint 3

```
1 class OrderedStream {  
2  
3     public OrderedStream(int n) {  
4  
5     }  
6  
7     public List<String> insert(int idKey, String value) {  
8  
9     }  
10 }  
11  
12 /**  
13  * Your OrderedStream object will be instantiated and called as  
14  * such:  
15  * List<String> param_1 = obj.insert(idKey,value);  
16  */
```

Problems

Pick One

< Prev

1656/2445

Next >

Console

Contribute i

Run Code ^

Submit