Snapshot Array

Try to solve the Snapshot Array problem.

We'll cover the following

Statement

In this challenge, you have to implement a **Snapshot Array** with the following properties:

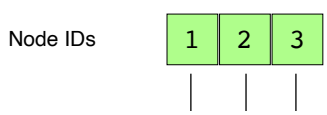
- **Constructor (length)**: This is the constructor and it initializes the data structure to hold the specified number of indexes.
- **Set Value (idx, val)**: This property sets the value at a given index **idx** to value **val**.
- **Snapshot()**: This method takes no parameters and returns the **Snap ID**. **Snap ID** is the number of times that the snapshot function was called, less 1, as we start the count at 0. The first time this function is called, it saves a snapshot and returns 0. The n^{th} time it is called, after saving the snapshot, it returns $n - 1$.
- **Get Value (idx, Snap ID)** method returns the value at the index in the snapshot with the given **Snap ID**.

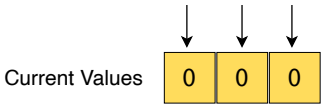
Suppose that we have three nodes whose values we wish to track in the snapshot array. Initially, the value of all the nodes will be 0. After calling the **Set Value (1, 4)** function, the value of node 1 will change to 4. If we take a snapshot at this point, the current values of all the nodes will be saved with **Snap ID** 0. Now, if we call **Set Value (1, 7)**, the current value for node 1 will change to 7. Now, if we call the **Get Value (1, 0)** function, we will get the value of node 1 from snapshot 0, that is, 4.

Constraints:

- $1 \leq$ `length` $\leq 5 \times 10^3$
- $0 \leq$ `idx` $<$ `length`
- $0 \leq$ `val` $\leq 10^9$
- $0 \leq$ `snapid` $<$ (the total number of times we call **Snapshot**)
- At most 5×10^3 calls will be made to **Set Value**, **Snapshot**, and **Get Value**.

Examples





The initial value for all nodes (1, 2, 3) is 0.

1 of 5

Understand the problem

Let's take a moment to make sure you've correctly understood the problem. The quiz below helps you check if you're solving the correct problem:

Snapshot Array

1 Select the correct output for the given code:

```
snapshot array = Constructor(3)snapshot array.set value(0, 4)snapshot array.snapshot()snapshot ar
ray.get value(0, 0)snapshot array.set value(1, 6) snapshot array.snapshot()snapshot array.get val
ue(1, 1)
```

A)
6
4

B)
4
6

C)
2
3

D)
3
2

Submit Answer

Question 1 of 2
0 attempted

Reset Quiz

Figure it out!

We have a game for you to play. Rearrange the logical building blocks to develop a clearer understanding of how to solve this problem.

Drag and drop the cards to rearrange them in the correct sequence.

Create a new entry in the dictionary by copying the previous value at `snapid` to `snapid + 1`.

Increment the value of `snapid` by 1 to keep track of the number of snapshots taken.

Return the index of the snapshot taken recently, which is the index at `snapid - 1`.

Reset | Show Solution | Submit

Try it yourself

Implement your solution in the following coding playground:

Java

usercode > SnapshotArray.java

```java
import java.util.*;
class SnapshotArray {

  // Constructor
  public SnapshotArray(int length) {
    // Write your code here
  }

  // Function set_value sets the value at a given index idx to val.
  public void setValue(int idx, int state) {
    // Write your code here
  }

  public int snapshot() {
    // Write your code here
    return -1;
  }
  // Function get_value returns the value at the index idx with the given snapid.
  public int getValue(int idx, int snapshotId1) {
    // Write your code here
    return -1;
  }
}

```

Powered by AI

Submit

Test Cases | Results

Case 1 | Case 2 | Case 3

Input #1

["SnapshotArray","snap"]

Input #2

[[4],[]]

Snapshot Array

Back

Custom Data Structur...

Next

Solution: Snapshot Ar...

Mark as Completed