



# Dynamic Array

Problem

Submissions

Leaderboard

Discussions

- Create a list, *seqList*, of  $N$  empty sequences, where each sequence is indexed from  $0$  to  $N - 1$ . The elements within each of the  $N$  sequences also use  $0$ -indexing.
- Create an integer, *lastAnswer*, and initialize it to  $0$ .
- The  $2$  types of queries that can be performed on your list of sequences (*seqList*) are described below:
  - Query:  $1 \ x \ y$ 
    - Find the sequence, *seq*, at index  $( (x \oplus \text{lastAnswer}) \% N )$  in *seqList*.
    - Append integer  $y$  to sequence *seq*.
  - Query:  $2 \ x \ y$ 
    - Find the sequence, *seq*, at index  $( (x \oplus \text{lastAnswer}) \% N )$  in *seqList*.
    - Find the value of element  $y \% \text{size}$  in *seq* (where *size* is the size of *seq*) and assign it to *lastAnswer*.
    - Print the new value of *lastAnswer* on a new line

## Task

Given  $N$ ,  $Q$ , and  $Q$  queries, execute each query.

**Note:**  $\oplus$  is the *bitwise XOR* operation, which corresponds to the  $\wedge$  operator in most languages. Learn more about it on [Wikipedia](#).

## Input Format

The first line contains two space-separated integers,  $N$  (the number of sequences) and  $Q$  (the number of queries), respectively. Each of the  $Q$  subsequent lines contains a query in the format defined above.

## Constraints

- $1 \leq N, Q \leq 10^5$
- $0 \leq x \leq 10^9$
- $0 \leq y \leq 10^9$
- It is guaranteed that query type  $2$  will never query an empty sequence or index.

## Output Format

For each type  $2$  query, print the updated value of *lastAnswer* on a new line.

## Sample Input

```
2 5
1 0 5
1 1 7
1 0 3
2 1 0
2 1 1
```

## Sample Output

7  
3

### Explanation

Initial Values:

$N = 2$

$lastAnswer = 0$

$S_0 = []$

$S_1 = []$

Query 0: Append **5** to sequence  $((0 \oplus 0) \% 2) = 0$ .

$lastAnswer = 0$

$S_0 = [5]$

$S_1 = []$

Query 1: Append **7** to sequence  $((1 \oplus 0) \% 2) = 1$ .

$S_0 = [5]$

$S_1 = [7]$

Query 2: Append **3** to sequence  $((0 \oplus 0) \% 2) = 0$ .

$lastAnswer = 0$

$S_0 = [5, 3]$

$S_1 = [7]$

Query 3: Assign the value at index **0** of sequence  $((1 \oplus 0) \% 2) = 1$  to  $lastAnswer$ , print  $lastAnswer$ .

$lastAnswer = 7$

$S_0 = [5, 3]$

$S_1 = [7]$

7

Query 4: Assign the value at index **1** of sequence  $((1 \oplus 7) \% 2) = 0$  to  $lastAnswer$ , print  $lastAnswer$ .

$lastAnswer = 3$

$S_0 = [5, 3]$

$S_1 = [7]$

3

f t in

Contest ends in a day

Submissions: 230



Max Score: 25




Difficulty: Easy

Rate This Challenge:

☆☆☆☆☆

More

Current Buffer (saved locally, editable)  

C++14   

```

1 #include <bits/stdc++.h>
2
3 using namespace std;
4
5 string ltrim(const string &);
6 string rtrim(const string &);
7 vector<string> split(const string &);
8
9 /* */
17
18 vector<int> dynamicArray(int n, vector<vector<int>> queries) {
19
20

```

```
21
22 int main()
23 {
68
69 string ltrim(const string &str) {
79
80 string rtrim(const string &str) {
90
91 vector<string> split(const string &str) {
107
```

Line: 1 Col: 1

 [Upload Code as File](#) ☐ Test against custom input

Run Code

Submit Code

[Contest Calendar](#) | [Interview Prep](#) | [Blog](#) | [Scoring](#) | [Environment](#) | [FAQ](#) | [About Us](#) | [Support](#) | [Careers](#) | [Terms Of Service](#) | [Privacy Policy](#) | [Request a Feature](#)