Dynamic Array ☆

RATE THIS CHALLENGE

☆ ☆ ☆ ☆ ☆

- 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:
  1. Query: `1 x y`
     1. Find the sequence, seq, at index $((x \oplus lastAnswer) \% N)$ in $seqList$.
     2. Append integer y to sequence seq.
  2. Query: `2 x y`
     1. Find the sequence, seq, at index $((x \oplus lastAnswer) \% N)$ in $seqList$.
     2. Find the value of element $y \% size$ in seq (where $size$ is the size of seq) and assign it to $lastAnswer$.
     3. 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 ^ 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
```

Java 8

```
16      public static List<Integer> dynamicArray(int n, List<List<Integer>> queries) {
17          final List<List<Integer>> tempList = new ArrayList<>();
18          for(int i = 0; i < n; i++) {
19              tempList.add(new ArrayList<>());
20          }
21          final List<Integer> result = new ArrayList<Integer>();
22          int lastAnswer = 0;
23
24          for (List<Integer> query : queries) {
25              int x = query.get(1);
26              int y = query.get(2);
27              int seqIndex = (x ^lastAnswer) % n;
28              if (query.get(0) == 1) {
29                  tempList.get(seqIndex).add(y);
30              } else {
31                  int index = y % tempList.get(seqIndex).size();
32                  lastAnswer = tempList.get(seqIndex).get(index);
33                  result.add(lastAnswer);
34              }
35          }
36          return result;
37      }
38
39      public static void main(String[] args) throws IOException {
40          BufferedReader bufferedReader = new BufferedReader(new InputStreamReader
    (System.in));
41          BufferedWriter bufferedWriter = new BufferedWriter(new FileWriter(System.getenv
    ("OUTPUT_PATH")));
42
```

Line: 1 Col: 1

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