

## Algorithm Class Mini-Contest 8

# Problem: JAMS

Time Limit: 2.0 seconds

Memory Limit: 64 MB

**Problem Description** There are  $N + 1$  cities in a row, numbered from 0 to  $N$ , with  $N$  roads running between them. Each road  $i$  connects cities  $i$  and  $i + 1$ . Each road  $i$  has a value  $P_i$  ( $2 \leq P_i \leq 6$ ), and the road is jammed from time  $kP_i$  to time  $kP_i + 1$ , where  $k$  is any non-negative integer.

When a road is jammed, no cars can pass through it, and has to wait at the city of origin until it stops being jammed, at which point it takes 1 time for the car to completely pass through the road to the next city.

Each *move* query asks you to consider the amount of time necessary for a car to move from city  $l$  to city  $r$  ( $l \leq r$ ), starting at time 0. Each *update* query can change the value of  $P_i$  for any road. There are  $Q$  queries in total.

**Input Format** The first line of input will contain two integers,  $N$  and  $Q$ . The next line of input will contain  $N$  integers, the array  $P$ . The next  $Q$  lines will contain one query each. If the first integer is 0, it is a *move* query, and the next two integers will be  $l$  and  $r$ . If the first integer is 1, it is an *update* query, and the next two integers will be  $i$  and  $P_i$ .

**Output Format** The output should contain one line with one integer for each *move* query, representing the minimum time needed to complete the task.

**Limits** These are the bounds on the input.

Subtask	Score	Additional Bounds
1	11	$1 \leq N, Q \leq 3,000$
2	23	$1 \leq N, Q \leq 100,000$ , no <i>update</i> queries.
3	28	$1 \leq N, Q \leq 100,000$ , $l = 0$ .
4	38	$1 \leq N, Q \leq 100,000$

### Sample Input

```
10 10
2 5 3 2 3 5 3 4 2 4
1 9 6
0 1 5
0 0 2
```

1 2 4  
0 2 10  
0 3 8  
0 4 5  
1 6 3  
0 7 9  
0 1 4

### Sample Output

5  
3  
14  
6  
2  
4  
4