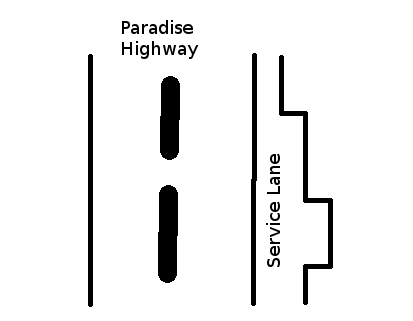
Problem: Service Lane

Calvin is driving his favorite vehicle on the 101 freeway. He notices that the check engine light of his vehicle is on, and he wants to service it immediately to avoid any risks. Luckily, a service lane runs parallel to the highway. The length of the service lane is  units. The service lane consists of  segments of equal length and different width.

Calvin can enter to and exit from any segment. Let's call the entry segment as index  and the exit segment as index . Assume that the exit segment lies after the entry segment () and . Calvin has to pass through *all*segments from index  to index  (both inclusive).



Calvin has three types of vehicles - bike, car, and truck - represented by ,  and , respectively. These numbers also denote the width of the vehicle.

You are given an array  of length , where  represents the width of the th segment of the service lane. It is guaranteed that while servicing he can pass through at most  segments, including the entry and exit segments.

* If , only the bike can pass through the th segment.
* If , the bike and the car can pass through the th segment.
* If , all three vehicles can pass through the th segment.

Given the entry and exit point of Calvin's vehicle in the service lane, output the type of the largest vehicle which can pass through the service lane (including the entry and exit segments).

**Input Format**

The first line of input contains two integers,  and , where  denotes the length of the freeway and  the number of test cases. The next line has  space-separated integers which represent the  array.

 test cases follow. Each test case contains two integers,  and , where  is the index of the segment through which Calvin enters the service lane and  is the index of the lane segment through which he exits.

**Constraints**

**Output Format**

For each test case, print the number that represents the largest vehicle type that can pass through the service lane.

*Note*: Calvin has to pass through *all* segments from index  to index  (both inclusive).

**Sample Input**

8 5

2 3 1 2 3 2 3 3

0 3

4 6

6 7

3 5

0 7

**Sample Output**

1

2

3

2

1

**Explanation**

Below is the representation of the lane:

|HIGHWAY|Lane| -> Width

0: | |--| 2

1: | |---| 3

2: | |-| 1

3: | |--| 2

4: | |---| 3

5: | |--| 2

6: | |---| 3

7: | |---| 3

1. (0, 3): Because *width[2] = 1*, only the bike can pass through it.
2. (4, 6): Here the largest allowed vehicle which can pass through the 5th segment is the car and for the 4th and 6thsegment it's the truck. Hence the largest vehicle allowed in these segments is a car.
3. (6, 7): In this example, the vehicle enters at the 6th segment and exits at the 7th segment. Both segments allow even trucks to pass through them. Hence the answer is *3*.
4. (3, 5): *width[3] = width[5] = 2*. While the 4th segment allows the truck, the 3rd and 5th allow up to a car. So *2* will be the answer here.
5. (0, 7): The bike is the only vehicle which can pass through the 2nd segment, which limits the strength of the whole lane to *1*.

Solution:

int checkLane(long entry, long exit, long array[])

{

int countCar=0;

for(int i=entry; i<=exit; i++)

{

if(array[i]==1)

{ return 1; }

else if(array[i]==2)

{countCar++;}

}

if(countCar>0) {return 2;}

return 3;

}

int main()

{

int testCases;

long width, entry, exit;

cin>>width >>testCases;

long array[width];

for(int i=0; i<width; i++)

{cin>>array[i];}

for(int i=0; i<testCases; i++)

{

cin>>entry >>exit;

cout<<checkLane(entry, exit, array)<<endl;

}

}

* Anshul Aggarwal