# 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 highway and the service lane is N units. The service lane consists of N segments of unit length, where each segment can have different widths.

Calvin can enter into and exit from any segment. Let's call the entry segment as index i and the exit segment as index j. Assume that the exit segment lies after the entry segment($j > i$) and $i \geq 0$. *Calvin has to pass through all segments from index i to index j (both inclusive).*
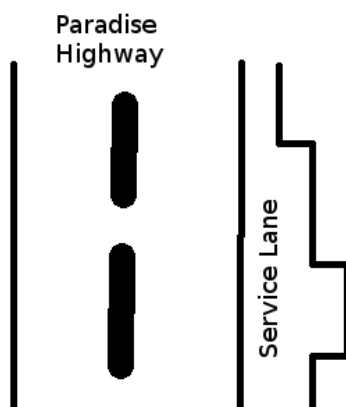
Calvin has three types of vehicles - bike, car and truck, represented by *1*, *2* and *3* respectively. These numbers also denote the width of the vehicle. We are given an array width[] of length N, where width[k] represents the width of kth segment of our service lane. It is guaranteed that while servicing he can pass through at most 1000 segments, including entry and exit segments.

- If width[k] is 1, only the bike can pass through k^{th} segment.

- If width[k] is 2, the bike and car can pass through k^{th} segment.

- If width[k] is 3, any of the bike, car or truck can pass through k^{th} segment.

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

Input Format
The first line of input contains two integers - N & T, where N is the length of the freeway, and T is the number of test cases. The next line has N space separated integers which represents the width array.

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

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 i to index j (both inclusive).

Constraints
2 <= N <= 100000
1 <= T <= 1000
0 <= i < j < N
2 <= j-i+1 <= min(N,1000)
1 <= width[k] <= 3, where 0 <= k < N

Sample Input #00

```
8 5
2 3 1 2 3 2 3 3
0 3
4 6
```

```
6 7
3 5
0 7
```

Sample Output #00

```
1
2
3
2
1
```

Explanation for Sample Case #0

Below is the representation of 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 represented as 1 can pass through it.

2. (4, 6): Here the largest allowed vehicle which can pass through the 5th segment is car and for the 4th and 6th segment 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 truck to pass through them. Hence truck is the answer.

4. (3, 5): $width[3] = width[5] = 2$. While 4th segment allow the truck, 3rd and 5th allow upto car. So 2 will be the answer here.

5. (0, 7): Bike is the only vehicle which can pass through the 2nd segment, which limits the strength of whole lane to 1.