

A non-empty zero-indexed array A of bits (1s and/or 0s) is given. The *maximal binary ones span* of A is the length of the longest sequence of consecutive 1s. For example, for A such that:

```
A[0] = 0
A[1] = 1
A[2] = 1
A[3] = 1
A[4] = 0
A[5] = 1
A[6] = 1
A[7] = 1
A[8] = 0
A[9] = 1
```

the maximal binary ones span equals 3. There are two sequences of consecutive 1s of such a length: A[1..3] and A[5..7].

You are given an implementation of a function:

```
int solution(vector<int> &A);
```

that, when given a non-empty array A consisting of N bits, finds the first sequence of consecutive 1s whose length equals the maximal binary ones span of A, and returns its starting position (index of the first 1 in the sequence).

If there are no bits set to 1, the function returns -1.

For example, given array A as in the example above, the function should return 1, because the maximal binary ones span in A equals 3 (which was already explained) and the first sequence of such length starts at index 1.

Given array A consisting of two bits such that:

```
A[0] = 0
A[1] = 0
```

the function should return -1, as there are no 1s.