Lesson 2, Task 2: Odd Occurrences in Array

A non-empty zero-indexed array A consisting of N integers is given. The array contains an odd number of elements, and each element of the array can be paired with another element that has the same value, except for one element that is left unpaired.

For example, in array A such that:

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

- the elements at indexes 0 and 2 have value 9,
- the elements at indexes 1 and 3 have value 3,
- the elements at indexes 4 and 6 have value 9,
- the element at index 5 has value 7 and is unpaired.

Write a function:

int solution(int A[], int N);

that, given an array A consisting of N integers fulfilling the above conditions, returns the value of the unpaired element.

For example, given array A such that:

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

the function should return 7, as explained in the example above.

Assume that:

- N is an odd integer within the range [1..1,000,000];
- each element of array A is an integer within the range [1..1,000,000,000];
- all but one of the values in A occur an even number of times.

Complexity:

- expected worst-case time complexity is O(N);
- expected worst-case space complexity is O(1), beyond input storage (not counting the storage required for input arguments).

Elements of input arrays can be modified.