codility

Candidate Report: Anonymous

Test Name:

Summary   Timeline

Test Score

100 out of 100 points

100%

Tasks in Test

	Time Spent ⓘ	Task Score
FrogRiverOne Submitted in: Python	4 min	100%

TASKS DETAILS

EASY

1. FrogRiverOne
Find the earliest time when a frog can jump to the other side of a river.

Task Score	Correctness	Performance
100%	100%	100%

Task description

A small frog wants to get to the other side of a river. The frog is initially located on one bank of the river (position 0) and wants to get to the opposite bank (position X+1). Leaves fall from a tree onto the surface of the river.

You are given an array A consisting of N integers representing the falling leaves. A[K] represents the position where one leaf falls at time K, measured in seconds.

The goal is to find the earliest time when the frog can jump to the other side of the river. The frog can cross only when leaves appear at every position across the river from 1 to X (that is, we want to find the earliest moment when all the positions from 1 to X are covered by leaves). You may assume that the speed of the current in the river is negligibly small, i.e. the leaves do not change their positions once they fall in the river.

For example, you are given integer X = 5 and array A such that:

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

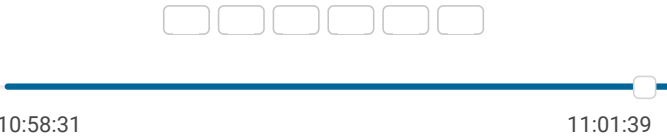In second 6, a leaf falls into position 5. This is the earliest time when leaves appear in every position across the river.

Write a function:

```
def solution(X, A)
```

Solution

Programming language used:	Python
Total time used:	4 minutes ❓
Effective time used:	4 minutes ❓
Notes:	*not defined yet*

Task timeline   ❓

☐ ☐ ☐ ☐ ☐ ☐

10:58:31   11:01:39

Code: 11:01:39 UTC, py, final,   show code in pop-up
score: **100**

```python
1   def solution(x, a):
2       way = set(range(1, x + 1))
3       path = set()
4       for i, leaf in enumerate(a):
5           path.add(leaf)
6           if path == way:
7               return i
8       return -1
```

that, given a non-empty array A consisting of N integers and integer X, returns the earliest time when the frog can jump to the other side of the river.

If the frog is never able to jump to the other side of the river, the function should return –1.

For example, given X = 5 and array A such that:

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

the function should return 6, as explained above.

Write an **efficient** algorithm for the following assumptions:

- N and X are integers within the range [1..100,000];
- each element of array A is an integer within the range [1..X].

Analysis summary

The solution obtained perfect score.

Analysis ❓

Detected time complexity:

O(N)

expand all	Example tests	
▶ example		✓ OK
example test		

expand all	Correctness tests	
▶ simple		✓ OK
simple test		
▶ single		✓ OK
single element		
▶ extreme_frog		✓ OK
frog never across the river		
▶ small_random1		✓ OK
3 random permutation, X = 50		
▶ small_random2		✓ OK
5 random permutation, X = 60		
▶ extreme_leaves		✓ OK
all leaves in the same place		

expand all	Performance tests	
▶ medium_random		✓ OK
6 and 2 random permutations, X = ~5,000		
▶ medium_range		✓ OK
arithmetic sequences, X = 5,000		
▶ large_random		✓ OK
10 and 100 random permutation, X = ~10,000		
▶ large_permutation		✓ OK
permutation tests		
▶ large_range		✓ OK
arithmetic sequences, X = 30,000		

PDF version of this report that may be downloaded on top of this site may contain sensitive data including personal information. For security purposes, we recommend you remove it from your system once reviewed.