Open Feedback Dialog

codility

Training center

ck out Codility training tasks

Congratulations

You have completed a Codility demo.

Tweet this!
I scored 100% in #py on @Codility!
http://codility.com/demo/take-sample-test/frog_river_one/

Sign up for our newsletter!

Like us on Facebook!

Demo ticket

Session

ID: demo6VZBKF-KQ7
Time limit: 120 min.

Status: closed

Created on: 2014-04-24 18:29 UTC
Started on: 2014-04-24 18:29 UTC
Finished on: 2014-04-24 19:59 UTC

Tasks in test		Correctness	Performance	Task score
1	{} FrogRiverOne	100%	100%	100%

Test score

?

100%

100 out of 100 points

EASY

1. FrogRiverOne

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

score: 100 of 100

Task description

A small frog wants to get to the other side of a river. The frog is currently located at position 0, and wants to get to position X. Leaves fall from a tree onto the surface of the river.

You are given a non-empty zero-indexed array A consisting of N integers representing the falling leaves. A[K] represents the position where one leaf falls at time K, measured in minutes.

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.

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 minute 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)
```

that, given a non-empty zero-indexed 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:

Solution

Programming language used: Python

Total time used: 91 minutes ?

Effective time used: 91 minutes ?

Notes: correct functionality and scalability

Task timeline ?

|◀ ◀◀ ▶ ■ ▶▶ ▶|

18:29:09 19:59:24

Code: 19:59:23 UTC, py, final, score: **100.00**

```
from collections import defaultdict
def solution(X, A):
    n = len(A)
    pos = defaultdict(int)
    for i in xrange(n):
        pos[A[i]] += 1
        if len(pos) == X:
            return i
    return -1
```

Analysis ?

```
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. Assume that:

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

Complexity:

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

Elements of input arrays can be modified.

Detected time complexity:

O(N)

test	time	result
Example tests		
example example test	0.050 s.	**OK**
Correctness tests		
simple simple test	0.050 s.	**OK**
single single element	0.050 s.	**OK**
extreme_frog frog never across the river	0.050 s.	**OK**
small_random1 3 random permutation, X = 50	0.050 s.	**OK**
small_random2 5 random permutation, X = 60	0.050 s.	**OK**
Performance tests		
medium_random 6 and 2 random permutations, X = ~5,000	0.070 s.	**OK**
medium_range arithmetic sequences, X = 5,000	0.060 s.	**OK**
large_random 10 and 100 random permutation, X = ~10,000	0.210 s.	**OK**
large_permutation permutation tests	0.300 s.	**OK**
large_range arithmetic sequences, X = 30,000	0.140 s.	**OK**
extreme_leaves all leaves in the same place	0.080 s.	**OK**

Training center