Practice Exercise #21: Rabbit Jumps

http://www.comp.nus.edu.sg/~cs1010/4_misc/practice.html

Reference: Week 5

Date of release: 8 September 2014

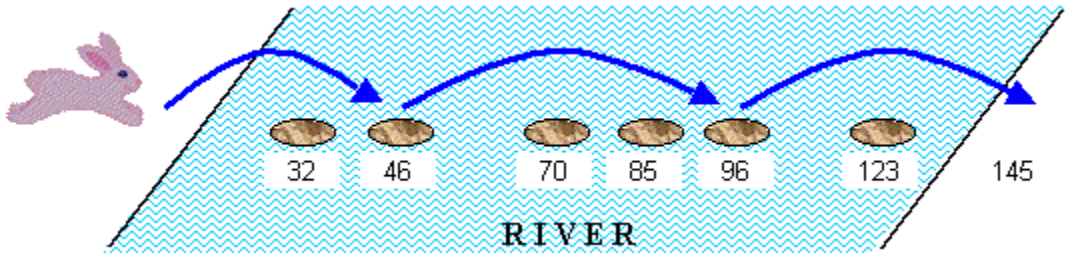
Objective: Array

Task statement:

(This is a past year's CS1101 Practical Exam question. The time limit given was 1 hour, so you should aim to complete it within an hour.)

A rabbit wants to cross to the other side of the river, but it cannot swim. However, it can hop, and each hop can bring it at most 50 cm away. So its only hope is to hop on the rocks, which are positioned in a straight line on the river. The positions of the rocks are measured from the start location, assuming to be zero. The opposite bank is treated as one big rock.

In the figure below, the rocks are at positions 32, 46, 70, 85, 96, 123, and the opposite riverbank at position 145.



The rabbit will jump as far as it could for each hop. **What is the smallest number of jumps it needs to take to reach the other side of the river?** For the above example, it is 3.

You may assume that there are at most 20 rocks (including the opposite bank).

Write a program **rabbit.c** that reads in a positive integer *n* that represents the number of rocks (including the opposite riverbank), and then on the next line *n* distinct positive integers in ascending order that represent the positions of the rocks. Your program should output the minimum number of jumps needed, or -1 if it is not possible for the rabbit to reach the other side of the river.

Your program should include a function

```
int countJumps(int [], int)
```

to compute the number of jumps required. The second parameter indicates the number of rocks in the array.

Sample runs:

```
Enter number of rocks: 7
32 46 70 85 96 123 145
3

Enter number of rocks: 5
40 70 150 160 180
-1

Enter number of rocks: 11
30 70 75 120 160 170 180 190 200 246 258
7
```