WhitePages Coding Challenge - ArrayHopper

Thanks again for your willingness to work on a coding question as part of the interview process for a Software Engineer position at WhitePages. The details you need for the coding challenge are below.

As a reminder, we'd like you to spend no more than four hours on this problem.

There is a strong possibility that you will have your solution coded well before that time, and that is perfectly fine. However, if you find that it is nearing the four-hour mark and you are not quite finished, please stop and send us what you have at that point. An incomplete solution is not necessarily the end of the interview process.

Please return the source code for your solution as an attachment in an email to me.If you have any questions about the instructions, feel free to contact me.

THE CHALLENGE

You are given an array of integers with values greater than or equal to 0, for example:

```
[5, 6, 0, 4, 2, 4, 1, 0, 0, 4]
```

You will develop and implement an algorithm to traverse the array in the shortest number of "hops" starting at index 0, where traversal is defined as follows:

- Start at the first (0th) index of the array, look at the array value there, and you can hop forward to any array index that is no farther than that value away. So in this example, you start at index 0 containing the value 5 and can now consider hopping to any of indices 1 through 5 inclusive.
- If you choose to hop to index 3, it contains the value 4 and you can next hop up to 4 more spots from your current index (3)—so you now consider indices 4 through 7 as next steps in your sequence.
- Once you can legally hop beyond the last array element, you have successfully traversed the array.

Your job is to compute the minimum-length sequence of hops that successfully traverses the array starting from index 0, or determine that there is no such sequence.

Your algorithm must identify a minimum-hops solution, but can choose arbitrarily among solutions with the same number of hops.

Your program will implement the algorithm and write a solution to the standard output stream on a single line identifying the array indices that comprise a solution path, with the indices separated by commas.

For this example, the following would be valid output:

```
0, 5, 9, out
```

(Note that your output is a sequence of array indices, not a sequence of hop lengths.)

- Your program may be written in the language of your choice.
- It must accept a single command-line argument, which is the path to a file containing the input data (the array of integers). Your program must not read that path from STDIN or the console.
- The input file will contain the array for a single problem.
  - One integer per line, with no brackets or commas.
  - The input array will likely be large in our testing, so performance is important.
- If there is no solution, the program should write the string "failure" to the standard output stream, followed by a newline character.