

Digit Chains

The Idea

Write a program in the language of your choice that finds the longest **chain** of integer strings from a library of strings without repeating a string.

The Goal

Given a library of unordered and unique integer strings between 1 and 20 digits each, identify a long chain of its members without repeating any. A **chain** is an ordered sequence of strings where adjacent strings differ by exactly one digit; either one digit has changed or one digit has been added or deleted. Your score will be the number of strings in the valid portion of your output chain, starting with the first string of your output. The higher the score, the better. For instance, given the library example from the next section, the following five output chains would score 5, 11, 4, 7 and 0, respectively:

Output chain	Score	Reason
54 4 2 5 15 52 32	5	15 and 52 are too different
49 4 0 08 02 32 2 52 5 54 84	11	Nothing is wrong with this
08 02 2 0 02 32 52 5 4	4	02 is repeated
02 08 0 4 2 5 15 154 54 84	7	154 doesn't exist
132 32 2 4 54 52	0	132 doesn't exist

Input

Your program should expect input to come into your program as if from a user at the keyboard, you can safely assume no malformed input will be provided.

- The first line indicates N, the number of digit strings in the library
- The only remaining line contains the N unique digit strings in the library, separated by space characters

Following is an example input stream for the 12 element digit string library used in previous examples:

12

4 2 15 0 32 49 02 5 54 84 08 52

Output

The output of your program must be a sequence of digit strings separated by single space characters. Once an invalid digit string, non-digit/space character or other unexpected output is encountered, the judging program will discard the unusual find along with all remaining output.

Evaluation

To evaluate your program, the judges will first compile your program (if your language requires compiling, that is). For the sake of example, let's assume you submit a source file called prog.c. The judges will perform the most basic compile unless you inform them more is required.

```
$ gcc -o prog prog.c
```

They will then execute the program using the following command (or similar, depending on the language and test file used):

```
$ cat 0.dat | ./prog | python3 ./judge.py 0.dat
```

This will provide the test data to your program as if it were being entered by someone at the console. It will also capture the output of your program and provide it to the judging program as if it were being entered by someone at the terminal. For your convenience, the judging program's source and data files used for judging are provided.

Runtime

While there is no official time limit for how long your program runs, it will be at the discretion of the judges to terminate your program whenever they see fit. Generally, we will let it run for a reasonably long time unless the testing machine gets too bogged down with concurrently running submissions.