S. A. Computer Olympiad

Second Round Open Div. 2007

For Grades 11, 12 and all wishing to qualify for the Final Round

Q1. Digit Sums

Proposed by Donald Cook

Prepared by Max Rabkin

Description

A simple operation that you can perform on a number is to sum its digits. If the result has more than 1 digit, the process may be repeated until a single digit answer is given. For example, applying the operation to 673 we get 7 i.e. 6 + 7 + 3 = 16 and 1 + 6 = 7.

Task

Your task is to write a program to calculate the digit sums described. The will consist of a number of lines, each containing a single positive number less than 100,000. The last line of input will contain 0 – this line should not be processed. The output will consist of a single line containing the digit sum (a single digit number) for each line of input, with no spaces separating them.

Sample run

Note the input is terminated by 0.

Input

```
673
 51
 1000
 99
 0
```

Output

```
7619
```

Test your program with

```
a.    26
      36
      37
      99
      0

b.    11111
      24567
      91919
      37459
      0

c.    1
      45
      3645
      99999
 0
```

Q2. Dominoes

Proposed by Marco Gallotta

Prepared by Shen Tian

Description

There are N dominoes arranged in a circle. They will be referred to as domino 0 through to N-1. The i^th^ domino has the value a_i on it.

Normally if you were to knock over one of the dominoes, this would cause the rest of the dominoes to knock each other over one by one until all the dominoes have been knocked over. However, these dominoes do not act as normal dominoes. These are magic dominoes. When the ith domino is knocked over, it tries to knock over domino $(i + a_i)$ mod n (where x mod y is equal to the remainder left when x is divided by y). However, if that domino has already been knocked over, the process ends and no more dominoes are knocked over. As a consequence, if we start with all the dominoes upright, we have to careful which domino we knock over. Only some dominoes, when knocked over, will start a chain reaction as described above that will knock over all the dominoes. We shall call these super dominoes.

Task

Write a program to determine, given N and all the < a_i s, how many super dominoes there are.

Sample run

N = 4, $a_0 = 1$, $a_1 = 3$, $a_2 = 5$, $a_3 = 1$

When you knock over domino 2, it knocks over domino 2+5 mod 4 = 3. 3+5 mod 4 = 0, so domino 0 is knocked over. 0+1 mod 4 = 1, thus domino 1, the last domino standing is knocked over. Domino 2 is a super domino.

If you knock over domino 1 however, domino 1+3 mod 4 = 0 is knocked over. Domino 0 tries to knock over domino 0+1 mod 4 = 1, which has already been knocked over. The process ends with dominoes 2 and 3 still standing.

One checks dominoes 0 and 3 similarly. Domino 2 is the only super domino in this case.

Input

A single integer N, followed by N integers for values of a_0 to a_{n-1}. Each integer has its own line.

Sample input:

```
 4
 1
 3
 5
 1
```

Output

Output the number of super dominoes

Sample output:

```
1
```

Test your program with

a. 5
1
1
1
1
1

b. 2
7
5

c. 3
54
1
1

d. 5
8
4
12
4
7

Q3. Hidden Strings

Proposed by Bruce Merry

Prepared by Ben Steenhuisen & Charles Bradshaw

Description

Strings are just a series of characters 'strung' or joined together. Substrings are strings that are, in fact, just a part of a larger string. One might, for various reasons, wish to find if a string is merely a substring of another string, sometimes disregarding such things as case (UPPER and lower) or punctuation.

Task

Your task is to write a program that finds and prints all occurrences of a word (substring) within a piece of text. This word may be hidden, it may contain spaces or punctuation, and it might appear with different capitalization. The program must accept 2 strings, the first being the main string, and the second the substring that is to be searched for in the main string. If no substrings are found, 'No strings found' must be printed.

Constraints

The length of each string will be <255 characters.

Sample run

Now that we are left with only a single letter, we are done.

Input

It's behind the intercom. Put erasers to one side
computer

Output

```
com. Put er
```

Test your program with

a. This suit is black!!
not

b. "You thought your secrets were safe. You were wrong."-Hackers
gh

c. Donald likes Mall shops where he and his friends discuss idealism all day long.
small

Q4. Tasks

Proposed by Bruce Merry

Prepared by Timothy Stranex

(This question is based on TopCoder SRM 337)

Description

You are playing a simple card game, in which the cards just have positive numbers N (which can be up to one million). There are also "jokers", which, for our purposes, have been allocated the number zero but can be used to represent any number. The goal of the game is to produce the longest "run" from the cards you have. A run is a set of cards showing consecutive numbers, for example 3, 4, 5, 6, 7 or 10, 11, 0, 13, 0 (with jokers standing in place of a 12 and a 14).

Task

Write a program that will take as input the cards in your hand and report the length of the longest run that can be made from it.

Constraints

$1 \le N \le 1\ 000\ 000$

Sample run

We have the cards 3 13 0 5 11 0 9 in our hand by inspection we can see that the longest run is 9, 0, 11, 0, 13 which is 5 long.

Input

Enter the numbers of the cards: 3 13 0 5 11 0 9

Output

The longest run has length 5.

Test your program with

a. 0, 0, 100, 0
b. 1, 0, 5, 0, 10, 0, 14, 11
c. 1000000, 500000, 499997, 0, 500002

Q5. Rooms

Proposed & Prepared by Marco Gallotta

Description

Fred the manic store-keeper can't keep up with the growing size of his store. He wants to know how many rooms he has and the size of the smallest and largest ones. However, his store is too large for him to work out on his own, so he has asked for your help.

Task

Fred has given you the plans of his store. In the plans, a wall is represented by a '1' and a floor tile by a '0'. Your task is to write a program to group neighbouring floor tiles into rooms. A tile can be grouped together with all tiles one space directly to its left, right, top and bottom. Note that this does *not* include diagonals. A room is defined as a group of floor tiles that cannot be grouped together with any further floor tiles.

Given the plans your task is to work out:

1. The total number of rooms
2. The size of the smallest room
3. The size of the largest room.

Constraints

1 <= width, height <= 20.

Sample run

Input

Enter width: 3
Enter height: 2
Enter row: 001
Enter row: 010

Output

Number of rooms: 2
 Smallest room: 1
 Largest room: 3

Test your program with

```
a.   01000
     11010
     00101

d.   010001000
     010101010
     010101010
     000100010

b.   00010101010100001000
     01010110101010010000
     01111100100011110000
     01000000110100001000
     01000010100100000101
     00100001000100000010
```
