Computer Olympiad
South African Computer Olympiad: a project of the Computer Society of South Africa.

Ph: 021-448 7864 • Fax: 021-447 8410 • PO Box 13013, MOWBRAY, 7705 • info@olympiad.org.za • www.olympiad.org.za

SECOND ROUND 2009

This paper is for ALL candidates

Each correct answer earns 5 marks

Q 1. Calculator Button

Authors: Michiel Baird and Robert Ketteringham
Proposed by: Max Rabkin

Introduction

Your calculator has a mysteriously unmarked button, that you suspect is either a plus (+) or a times (x). If you press the button between two numbers, and examine the calculator's output, you could find out what the button could be.

Task

Write a program that determines which button it can be: a plus (+) or times (x) button. Your program should read in three numbers: the first two being the numbers entered into the calculator along with the mysterious button and the last number, being the calculator's answer.

If the button can be either a plus *or* times button, output "Plus or Times". If the button can *only* be a plus or *only* be a times button, output which one it can be ("Plus only" or "Times only"). If it can be neither plus *nor* times, output "Neither Plus nor Times". Lastly, output both the plus and times equations.

Sample Run

Input

```
First number: 4
Second number: 2
Answer: 6
```

Output

```
Plus only
4+2=6
4x2=8
```

Test Your Program With

(i)	5	6	11	(5)
(ii)	2	4	8	(5)
(iii)	2	2	4	(5)
(iv)	7	2	4	(5)

Q 2. Bar Graph

Authors: Donald Cook and Peter Waker

Introduction

A bar graph represents numbers by the length of either a vertical or horizontal bar."

Task

Write a program to create a bar graph that will represent any number as a series of horizontal bars, representing the digits of the number.

Your program must first print the digit and then the bar, using asterisks (*), as in the example given below. Only positive integers will be given, and no number will have more than 9 digits.

Sample Run

Input

```
Enter the number: 5103
```

Output

```
5 *****
1 *
0
3 ***
```

Test Your Program With

(i)	4	(5)
(ii)	429	(5)
(iii)	4870864	(5)
(iv)	123456789	(5)

Supported by Mark Shuttleworth, the University of Cape Town and the Centre for High Performance Computing.

Sponsored by Standard Bank

Q 3. Arithmetic Progression

Author: Julian Kenwood
Proposed by: Bruce Merry

Introduction

An arithmetic progression is a sequence where the difference between any two consecutive numbers is always the same. This difference is known as the common difference. In an increasing arithmetic progression this common difference is greater than zero.

Task

Input N distinct integers. Your task is to find the longest increasing arithmetic progression in these numbers. Note that there could be multiple longest increasing arithmetic progressions. First output the length of the longest arithmetic progression. Then output the number of increasing arithmetic progressions with this length. Finally output the first term and the common difference of the longest increasing arithmetic progression with the smallest first term. If more than one of them has that first term, output the smallest common difference.

Sample Run

Input

```
Enter N: 5
Enter a number: 1
Enter a number: 5
Enter a number: 9
Enter a number: 11
Enter a number: 3
```

Output

```
Arithmetic Progression Length = 3
Number of Arithmetic Progressions = 2
Arithmetic Progression First Term = 1
Arithmetic Progression Common
Difference = 2
```

Explanation

There are two increasing arithmetic progressions of length 3 in the set {1, 5, 9, 11, 3}:

- 1 3 5, with first term 1 and a common difference of 2
- 1 5 9, with first term 1 and a common difference of 4

Test Your Program With

(i) N = 5
1 5 3 9 7 (5)

(ii) N=10
103 304 2004 19 35 85 13 78 10 77
(5)

(iii) N=10
1 4 9 16 25 36 49 64 81 100 (5)

(iv) N=15
1 15 4 7 11 10 8 9 13 16 19 23 22 26 29
(5)

Q 4. Robot Moves

Authors: Alex van Olst and Henk Joubert
Proposed by: Bruce Merry

Introduction

A robot is sitting somewhere in an MxN grid of cells (M columns, N rows). It receives and interprets a string of instructions. The available instructions are u (up), d (down), l (left) and r (right). If a robot can move in the direction specified, it moves one cell in this direction. However, if it is already at the boundary, it ignores the instruction.

Task

A known string of instructions has been sent to the robot, but since you don't know where it started, you have no idea where it has moved to. You can issue some further instructions that will move it to a precise location that is independent of the robot's starting position. Write a program that will work out the fewest extra instructions required to move it to such position.

Sample Run

The robot starts in a unknown position in a grid of 5 columns and 4 rows. The robot receives an instruction list of: left, up, left, down, right, down. The robot then attempts to follow the instructions ignoring those that would result in it leaving the grid. If you then send the robot 4 additional instructions (down, right, right, right) then it will be in the bottom right corner regardless of where it started.

Input

```
Enter the width M: 5
Enter the height N: 4
Enter the initial instructions: luldrd
```

Output

```
You need to issue a further 4
instructions.
```

Test Your Program With

(i) 1 1 lrudlrudlu (5)

(ii) 100 100 lrudrudludlrdlru (5)

(iii) 32 16
llllllllllllllllllllllllllllllluuuuuuuuuuuuuuuuddddddddddddddddd [That is, 30***l***, 16***u***, 16***d***] (5)

(iv) 1000 3 ulrd (5)

Q 5. Arithmetic Game

Author: Max Rabkin

Introduction

Migael is playing an arithmetic game, and wants your help practising. The game is to make as many numbers as possible with multiplication and addition, using numbers from a given list. In order to check his work, Migael wants to know how many possible numbers can be formed, and what their sum is.

Task

Given a list of numbers, find all the answers that can be formed from that list using addition and multiplication. In calculating each answer, each number must be used exactly as many times as it appears in the input. Output the number of different answers and the sum of all the answers (do not count an answer twice even if it can be formed in two different ways).

Sample Run

Input

```
What numbers do you have? 1 2 3
```

Output

```
You can form 5 numbers
Their sum is 35
```

The numbers which can be formed are:
1*2+3=1*3+2=**5**, 1+2+3=1*2*3=**6**,
(2*3)+1=**7**, 2*(3+1)=**8**, (2+1)*3=**9**

Test Your Program With

(i) 1 2 4 (5)

(ii) 3 4 5 (5)

(iii) 3 19 23 88 (5)

(iv) 15 37 26 1001 32 (5)

[20 x 5 = 100]