ADPF Assignment – 1

1. You will be given 3 integers as input. The inputs may or may not be different from each other.
   You have to output 1 if all three inputs are different from each other, and 0 if any input is repeated more than once.

   Input-----
   Three integers on three lines.

   Output------
   1 if the three inputs are different from each other
   , 0 if some input is repeated more than once

Private Test cases used for evaluation	**Input**	**Expected Output**
Test Case 1	100 5 100	0
Test Case 2	50 25 25	0
Test Case 3	1 6 1	0
Test Case 4	4 7 0	1

2. You are given two integers, say M and N.

   You must check whether M is an exact multiple of N, without using loops.
   You have to output 0 if M is not a multiple of N.
   You have to output M/N if M is a multiple of N.

   Input-----
   Two integers, say M and N.
   Output------
   You have to output 0 if M is not a multiple of N.
   You have to output M/N if M is a multiple of N.

Private Test cases used for evaluation	**Input**	**Expected Output**
Test Case 1	100 5	20
Test Case 2	16 4	4
Test Case 3	4 16	0
Test Case 4	45 49	0

3. Input : Triplet of three numbers (a,b,c)
   Output : 1 if a>b>c
   0, otherwise

Private Test cases used for evaluation	Input	Expected Output
Test Case 1	1 4 6	0
Test Case 2	1 1 0	0
Test Case 3	1 -1 -4	1

4. You are given a **sorted** (either in the increasing or in the decreasing order) sequence of numbers, ending with a -1. You can assume that are at least two numbers before the ending -1.
   Let us call the sequence x0 x1 ... xn -1.
   You have to output the number of distinct elements in the sorted sequence.

   Kindly do not use arrays in the code.

Sample Test Cases	Input	Output
Test Case 1	`5 4 4 3 -1`	3
Test Case 2	`1 2 -1`	2
Test Case 3	`12 33  87 87 87 87 87  94 112 120 -1`	6
Test Case 4	`1 2 2 -1`	2
Test Case 5	`4 5 6 7 -1`	4

5. You are given a sequence of non-negative integers terminated by -1. You have to output 1 if there are at least 2 distinct elements in the sequence and 0 if the sequence consists of only 1 integer. Note that -1 is not part of the sequence. The sequence is not necessarily sorted.

   Note: Don't use arrays to this question.

Sample Test Cases	**Input**	**Output**
Test Case 1	`1 1 1 1 -1`	`0`
Test Case 2	`1 2 3 4 -1`	`1`
Test Case 3	`4 5 3 5 -1`	`1`
Test Case 4	`4 4 4 4 4 -1`	`0`
Test Case 5	`1 1 1 2 -1`	`1`

6. The length of three sides are taken as input. Write a C program to find whether a triangle can be formed or not.

   If not, display “This Triangle is NOT possible.” If the triangle can be formed then check whether the triangle formed is equilateral, isosceles, scalene or a right-angled triangle.

   (If it is a right-angled triangle then only print Right-angle triangle do not print it as Scalene Triangle).

Test cases	**Input**	**Expected Output**
Test Case 1	5 12 13	Right-angle Triangle
Test Case 2	9 9 9	Equilateral Triangle

7. Write a program to find the factorial of a given number using while loop.

Test cases	**Input**	**Expected Output**
Test Case 1	7	The Factorial of 7 is : 5040
Test Case 2	11	The Factorial of 11 is : 39916800

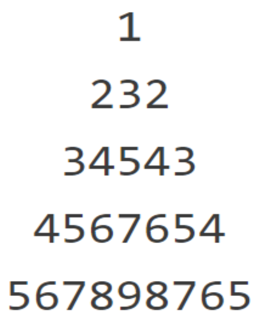
8. Write a C program to calculate the Sum of First and the Last Digit of a given Number. For example if the number is 1234 the result is 1+4 = 5.

Test cases	**Input**	**Expected Output**
Test Case 1	2001	Sum of First and Last digit = 3
Test Case 2	45632	Sum of First and Last digit = 6

9. Write a program in C to display the pattern like a diamond.

```
    *
   * * *
  * * * * *
 * * * * * * *
* * * * * * * * *
 * * * * * * *
  * * * * *
   * * *
    *
```

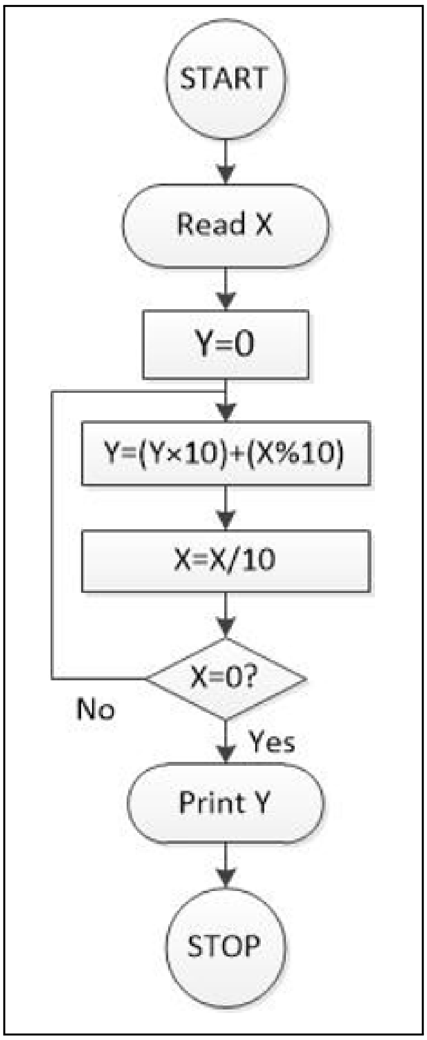
10. Write Program to generate following pattern for input size N.



```
    1
   232
  34543
 4567654
567898765
```

For N=5 output is:

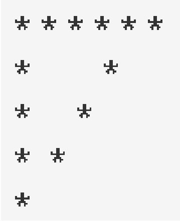
11. Create a C program for the following flow chart. What will be the output if the user enters X = 2648 ?



12. Create a algorithm and flowchart for multiplying two integer numbers without using multiplication operator (*). Write a C program for implementing the same algorithm.

13. Write a C program to generate the following pattern.

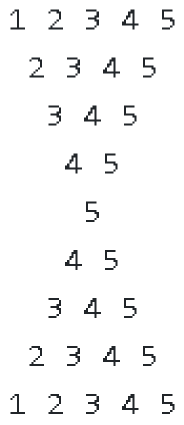
E.g. for N=5



```
* * * * * *
*      *
*    *
*  *
*
```

14. Write a C program to generate the following pattern.

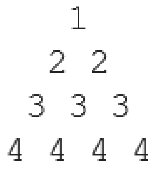
E.g. for N=5



```
1 2 3 4 5
 2 3 4 5
  3 4 5
   4 5
    5
   4 5
  3 4 5
 2 3 4 5
1 2 3 4 5
```

15. Write a program in C to make the pattern below.

E.g. for N=4



```
   1
  2 2
 3 3 3
4 4 4 4
```

16. Write a C program to print a well formatted multiplication table of any number.