**RECURSION ASSIGNMENT**

Q1. Multiplication (Recursive)

**Given two integers M & N, calculate and return their multiplication using recursion. You can only use subtraction and addition for your calculation. No other operators are allowed.**

**Input format :**

Line 1 : Integer M

Line 2 : Integer N

**Output format :**

M x N

**Constraints :**

0 <= M <= 1000

0 <= N <= 1000

**Sample Input 1 :**

3

5

**Sample Output 1 :**

15

**Sample Input 2 :**

4

0

**Sample Output 2 :**

0

Q2. Print Numbers

**Given the number 'n', write a code to print numbers from 1 to n in increasing order recursively.**

**Input Format :**

Integer n

**Output Format :**

Numbers from 1 to n (separated by space)

**Constraints :**

1 <= n <= 10000

**Sample Input 1 :**

6

**Sample Output 1 :**

1 2 3 4 5 6

**Sample Input 2 :**

4

**Sample Output 2 :**

1 2 3 4

Q3. Number of Digits

**Given the number 'n', find out and return the number of digits present in a number .**

**Input Format :**

Integer n

**Output Format :**

Count of digits

**Constraints :**

1 <= n <= 10^6

**Sample Input 1 :**

156

**Sample Output 1 :**

3

**Sample Input 2 :**

7

**Sample Output 2 :**

1

Q4. Sum of Natural Numbers

**Given an integer n, find and return the sum of numbers from 1 to n using recursion.**

**Input Format :**

Integer n

**Output Format :**

Sum

**Constraints :**

**1 <= n <= 10^3**

**Sample Input :**

10

**Sample Output :**

55

Q5. Sum of Array

**Given an array of length N, you need to find and return the sum of all elements of the array.**

**Do this recursively.**

**Input Format :**

Line 1 : An Integer N i.e. size of array

Line 2 : N integers which are elements of the array, separated by spaces

**Output Format :**

Sum

**Constraints :**

1 <= N <= 10^3

**Sample Input 1 :**

3

9 8 9

**Sample Output 1 :**

26

**Sample Input 2 :**

3

4 2 1

**Sample Output 2 :**

7

Q6. Check Number in Array

**Given an array of length N and an integer x, you need to find if x is present in the array or not. Return true or false.**

**Do this recursively.**

**Input Format :**

Line 1 : An Integer N i.e. size of array

Line 2 : N integers which are elements of the array, separated by spaces

Line 3 : Integer x

**Output Format :**

'true' or 'false'

**Constraints :**

1 <= N <= 10^3

**Sample Input 1 :**

3

9 8 10

8

**Sample Output 1 :**

true

**Sample Input 2 :**

3

9 8 10

2

**Sample Output 2 :**

false

Q7. First Index of Number

**Given an array of length N and an integer x, you need to find and return the first index of integer x present in the array. Return -1 if it is not present in the array.**

**First index means, the index of first occurrence of x in the input array.**

**Do this recursively. Indexing in the array starts from 0.**

**Input Format :**

Line 1 : An Integer N i.e. size of array

Line 2 : N integers which are elements of the array, separated by spaces

Line 3 : Integer x

**Output Format :**

first index or -1

**Constraints :**

**1 <= N <= 10^3**

**Sample Input :**

4

9 8 10 8

8

**Sample Output :**

1

Q8. Count Zeros

**Given an integer N, count and return the number of zeros that are present in the given integer using recursion.**

**Input Format :**

Integer N

**Output Format :**

Number of zeros in N

**Constraints :**

0 <= N <= 10^9

**Sample Input 1 :**

0

**Sample Output 1 :**

1

**Sample Input 2 :**

00010204

**Sample Output 2 :**

2

**Explanation for Sample Output 2 :**

Even though "00010204" has 5 zeros, the output would still be 2 because when you convert it to an integer, it becomes 10204.

**Sample Input 3 :**

708000

**Sample Output 3 :**

4

Q9. Check Palindrome (recursive)

**Check whether a given String S is a palindrome using recursion. Return true or false.**

**Input Format :**

String S

**Output Format :**

'true' or 'false'

**Constraints :**

0 <= |S| <= 1000

where |S| represents length of string S.

**Sample Input 1 :**

facecaf

**Sample Output 1:**

true

**Sample Input 2 :**

chitkara

**Sample Output 2:**

false

Q10. Sum of digits (recursive)

**Write a recursive function that returns the sum of the digits of a given integer.**

**Input format :**

Integer N

**Output format :**

Sum of digits of N

**Constraints :**

0 <= N <= 10^9

**Sample Input 1 :**

12345

**Sample Output 1 :**

15

**Sample Input 2 :**

9

**Sample Output 2 :**

9

Q11. Tower of Hanoi

**Tower of Hanoi is a mathematical puzzle where we have three rods and n disks. The objective of the puzzle is to move all disks from source rod to destination rod using third rod (say auxiliary). The rules are :**

1) Only one disk can be moved at a time.

2) A disk can be moved only if it is on the top of a rod.

3) No disk can be placed on the top of a smaller disk.

**Print the steps required to move n disks from source rod to destination rod.**

**Source Rod is named as 'a', auxiliary rod as 'b' and destination rod as 'c'.**

**Input Format :**

Integer n

**Output Format :**

Steps in different lines (in one line print source and destination rod name separated by space)

**Constraints :**

0 <= n <= 20

**Sample Input 1 :**

2

**Sample Output 1 :**

a b

a c

b c

**Sample Input 2 :**

3

**Sample Output 2 :**

a c

a b

c b

a c

b a

b c

a c

Q12. String to Integer

**Write a recursive function to convert a given string into the number it represents. That is input will be a numeric string that contains only numbers, you need to convert the string into corresponding integer and return the answer.**

**Input format :**

Numeric string S (string, Eg. "1234")

**Output format :**

Corresponding integer N (int, Eg. 1234)

**Constraints :**

0 < |S| <= 9

where |S| represents length of string S.

**Sample Input 1 :**

00001231

**Sample Output 1 :**

1231

**Sample Input 2 :**

12567

**Sample Output 2 :**

12567

Q13. Pair Star

**Given a string S, compute recursively a new string where identical chars that are adjacent in the original string are separated from each other by a "\*".**

**Input format :**

String S

**Output format :**

Modified string

**Constraints :**

0 <= |S| <= 1000

where |S| represents length of string S.

**Sample Input 1 :**

hello

**Sample Output 1:**

hel\*lo

**Sample Input 2 :**

aaaa

**Sample Output 2 :**

a\*a\*a\*a

Q14. Binary Search (Recursive)

**Given an integer sorted array (sorted in increasing order) and an element x, find the x in given array using binary search. Return the index of x.**

**Return -1 if x is not present in the given array.**

**Note : If given array size is even, take first mid.**

Input format :

Line 1 : Array size

Line 2 : Array elements (separated by space)

Line 3 : x (element to be searched)

**Sample Input :**

6

2 3 4 5 6 8

5

**Sample Output:**

3

Q15. Staircase

**A child is running up a staircase with N steps, and can hop either 1 step, 2 steps or 3 steps at a time. Implement a method to count how many possible ways the child can run up to the stairs. You need to return number of possible ways W.**

**Input format :**

Integer N

**Output Format :**

Integer W

**Constraints :**

1 <= N <= 30

**Sample Input 1 :**

4

**Sample Output 1 :**

7

**Sample Input 2 :**

5

**Sample Output 2 :**

13

Chocolate question:

Print Permutations

**Given an input string (STR), print all possible permutations of the input string.**

**Note:**

The input string may contain the same characters, so there will also be the same permutations.

The order of permutations doesn’t matter.

**Input Format:**

The only input line contains a string (STR) of alphabets in lower case

**Output Format:**

Print each permutations in a new line

**Constraint:**

1<=length of STR<=8

Time Limit: 1sec

**Sample Input 1:**

cba

**Sample Output 1:**

abc

acb

bac

bca

cab

cba