SUM OF ODD PLACED AND EVEN PLACED DIGITS BASIC

Take N as input. Print the sum of its odd placed digits and sum of its even placed digits.

**Constraints:**

0 < N <= 1000000000

**Sample Input:**

2635

**Sample Output:**

11

5

**Explanation:**

5 is present at 1st position, 3 is present at 2nd position, 6 is present at 3rd position and 2 is present at 4th position.

Sum of odd placed digits on first line. 5 and 6 are placed at odd position. Hence odd place sum is 5+6=11

Sum of even placed digits on second line. 3 and 2 are placed at even position. Hence even place sum is 3+2=5

INCREASING DECREASING SEQUENCE BASICS

Take the following as input.

A number (N)  
Take N more numerical inputs  
The N inputs for a sequence S = s1, s2, .., sN. Compute if it is possible to split sequence into two sequences -  
s1 to si and si+1 to sN such that first sequence is strictly decreasing and second is strictly increasing. Print true/false as output.

**Constraints:**

0 < N < 1000

Each number in sequence S is > 0 and < 1000000000

**Sample Input:**

5

1

2

3

4

5

**Sample Output:**

true

**Explanation:**

Carefully read the conditions to judge which all sequences may be valid. Don't use arrays or lists.

COUNT DIGITS BASICS

Take the following as input.  
A number  
A digit  
Write a function that returns the number of times digit is found in the number. Print the value returned.

**Input Format:**

Integer (A number) Integer (A digit)

**Constraints:**

0 <= N <= 1000000000

0 <= Digit <= 9

**Output Format:**

Integer (count of times digit occurs in the number)

**Sample Input:**

5433231

3

**Sample Output:**

3

**Explanation:**

The digit can be from 0 to 9. Assume decimal numbers

PRINT SERIES BASICS

Take the following as input.

A number (N1)  
A number (N2)  
Write a function which prints first N1 terms of the series 3n + 2 which are not multiples of N2.

**Constraints:**

0 < N1 < 100

0 < N2 < 100

**Sample Input:**

10

4

**Sample Output:**

5

11

14

17

23

26

29

35

38

41

**Explanation:**

The output will've N1 terms.

ARRAYS-REVERSE AN ARRAY ARRAYS

Take as input N, the size of array. Take N more inputs and store that in an array. Write a function that reverses the array. Print the values in reversed array.

1.It reads a number N.  
2.Take Another N numbers as input and store them in an Array.  
3.Reverse the elements in the Array.  
4.Print the reversed Array.

**Constraints:**

N cannot be Negative. Range of Numbers can be between -1000000000 to 1000000000.

**Sample Input:**

5

0

4

6

8

9

**Sample Output:**

9

8

6

4

0

**Explanation:**

In the sample case , arr=[0,4,6,8,9] is reversed to arr=[9,8,6,4,0].

ARRAYS-TARGET SUM PAIRS ARRAYS

Take as input N, the size of array. Take N more inputs and store that in an array. Take as input “target”, a number. Write a function which prints all pairs of numbers which sum to target.

**Constraints:**

Length of the arrays should be between 1 and 1000.

**Sample Input:**

5

1

3

4

2

5

5

**Sample Output:**

1 and 4

2 and 3

**Explanation:**

Find any pair of elements in the array which has sum equal to target element and print them.

ARRAYS-WAVE PRINT ROW WISE ARRAYS

Take as input a two-d array. Wave print it row-wise.

**Input Format:**

Two integers M(row) and N(column) and further M \* N integers(2-d array numbers).

**Constraints:**

Both M and N are between 1 to 10.

**Output Format:**

All M \* N integers are seperated by commas with 'END' written in the end(as shown in example).

**Sample Input:**

4 4

11 12 13 14

21 22 23 24

31 32 33 34

41 42 43 44

**Sample Output:**

11, 12, 13, 14, 24, 23, 22, 21, 31, 32, 33, 34, 44, 43, 42, 41, END

ARRAYS-WAVE PRINT COLUMN WISE ARRAYS

Take as input a two-d array. Wave print it column-wise.

**Input Format:**

Two integers M(row) and N(colomn) and further M \* N integers(2-d array numbers).

**Constraints:**

Both M and N are between 1 to 10.

**Output Format:**

All M \* N integers seperated by commas with 'END' wriiten in the end(as shown in example).

**Sample Input:**

4 4

11 12 13 14

21 22 23 24

31 32 33 34

41 42 43 44

**Sample Output:**

11, 21, 31, 41, 42, 32, 22, 12, 13, 23, 33, 43, 44, 34, 24, 14, END

ARRAYS-SPIRAL PRINT CLOCKWISE ARRAYS

Take as input a two-d array. Spiral print it clockwise.

**Input Format:**

Two integers M(row) and N(colomn) and further M \* N integers(2-d array numbers).

**Constraints:**

Both M and N are between 1 to 10.

**Output Format:**

All M \* N integers seperated by commas with 'END' wriiten in the end(as shown in example).

**Sample Input:**

4 4

11 12 13 14

21 22 23 24

31 32 33 34

41 42 43 44

**Sample Output:**

11, 12, 13, 14, 24, 34, 44, 43, 42, 41, 31, 21, 22, 23, 33, 32, END

**Explanation:**

For spiral level clockwise traversal, Go for first row-> last column ->last row -> first column and then do this for the same remaining matrix. Handle to print same elements multiple time.

SORTING IN LINEAR TIME ARRAYS

You will be given an array containing only 0s, 1s and 2s. you have sort the array in linear time that is O(N) where N is the size of the array.

**Input Format:**

The first line contains N, which is the size of the array. The following N lines contain either 0, or 1, or 2.

**Constraints:**

Each input element x, such that x ∈ { 0, 1, 2 }.

**Output Format:**

Output the sorted array with each element separated by a newline.

**Sample Input:**

5

0

1

2

1

2

**Sample Output:**

0

1

1

2

2

ARRAYS-INTERSECTION OF TWO ARRAYS ARRAYS

Take as input N, the size of array. Take N more inputs and store that in an array. Take N more inputs and store that in another array. Write a function which gives the intersection of two arrays in an List of integers and Print the List.

**Constraints:**

Length of Arrays should be between 1 to 1000.

**Sample Input:**

7

1 2 3 1 2 4 1

2 1 3 1 5 2 2

**Sample Output:**

[1, 1, 2, 2, 3]

**Explanation:**

Check which integer is present in both the arrays and add them in an list .Print List as the ans.

ARRAYS-SUM OF TWO ARRAYS ARRAYS

Take as input N, the size of array. Take N more inputs and store that in an array. Take as input M, the size of second array and take M more inputs and store that in second array. Write a function that returns the sum of two arrays. Print the value returned.

**Constraints:**

Length of Array should be between 1 and 1000.

**Sample Input:**

4

1 0 2 9

5

3 4 5 6 7

**Sample Output:**

3, 5, 5, 9, 6, END

**Explanation:**

Sum of [1, 0, 2, 9] and [3, 4, 5, 6, 7] is [3, 5, 5, 9, 6] and the first digit represents carry over , if any (0 here ) .

STRINGS-COUNT PALINDROMIC SUBSTRINGS STRINGS

Take as input S, a string. Write a program that gives the count of substrings of this string which are palindromes and Print the ans.

**Input Format:**

String

**Constraints:**

Length of string is between 1 to 1000.

**Output Format:**

Integer

**Sample Input:**

abc

**Sample Output:**

3

**Explanation:**

For the given sample case , the palindromic substrings of the string abc are "a","b" and "c".So, the ans is 3.

STRINGS-ODD EVEN CHARACTER STRINGS

Take as input S, a string. Write a function that replaces every odd character with the character having just higher ascii code and every even character with the character having just lower ascii code. Print the value returned.

**Input Format:**

String

**Constraints:**

Length of string should be between 1 to 1000.

**Output Format:**

String

**Sample Input:**

abcg

**Sample Output:**

badf

STRINGS-MAX FREQUENCY CHARACTER STRINGS

Take as input S, a string. Write a function that returns the character with maximum frequency. Print the value returned.

**Input Format:**

String

**Constraints:**

A string of length between 1 to 1000.

**Output Format:**

Character

**Sample Input:**

aaabacb

**Sample Output:**

a

**Explanation:**

For the given input string, a appear 4 times. Hence, it is the most frequent character.

STRINGS-REMOVE DUPLICATES STRINGS

Take as input S, a string. Write a function that removes all consecutive duplicates. Print the value returned.

**Input Format:**

String

**Constraints:**

A string of length between 1 to 1000

**Output Format:**

String

**Sample Input:**

aabccba

**Sample Output:**

abcba

**Explanation:**

For the given example, "aabccba", Consecutive Occurrence of a is 2, b is 1, and c is 2.

After removing all of the consecutive occurences, the Final ans will be : - "abcba".

STRINGS-STRING COMPRESSION STRINGS

Take as input S, a string. Write a function that does basic string compression. Print the value returned. E.g. for input “aaabbccds” print out a3b2c2ds.

**Input Format:**

A single String S.

**Constraints:**

A string of length between 1 to 1000

**Output Format:**

The compressed String.

**Sample Input:**

aaabbccds

**Sample Output:**

a3b2c2ds

**Explanation:**

In the given sample test case 'a' is repeated 3 times consecutively, 'b' is repeated twice, 'c' is repeated twice. But, 'd' and 's' occurred only once that's why we do not write their occurrence.

CALCULATE THE SUM ARRAYS

Raj is a very smart kid who recently started learning computer programming. His coach gave him a cyclic array A having N numbers, and he has to perform Q operations on this array. In each operation the coach would provide him with a number X. After each operation, every element of the cyclic array would be replaced by the sum of itself and the element lying X positions behind it in the cyclic array. All these replacements take place simultaneously. For example, if the cyclic array was [a, b, c, d], then after the operation with X = 1, the new array would be [a+d, b+a, c+b, d+c]. He needs to output the sum of the elements of the final array modulus 10^9+7. He made a program for it but it's not very efficient. You know he is a beginner, so he wants you to make an efficient program for this task because he doesn't want to disappoint his coach.

**Input Format:**

The first line of each test file contains a integer N. The next line contains N space separated integers which represent the elements of the cyclic array. The third line contains a integer Q representing the number of operations that will be applied to the array. Finally, Q lines follow, each one containing an integer X .

**Constraints:**

1 <= N <= 100000

1 <= Ai<= 10^9

0 <= Q <= 1000000

0 <= X < N

**Output Format:**

Your program should output to the standard output stream the sum of the elements of the final array modulus 10^9+7.

**Sample Input:**

5

1 2 3 4 5

2

1

0

**Sample Output:**

60

**Explanation:**

After the 1st operation (X = 1), the array would be [1+5, 2+1, 3+2, 4+3, 5+4] =[6, 3, 5, 7, 9]  
After 2nd operation (X = 0), the array would be [6+6, 3+3, 5+5, 7+7, 9+9] =[12, 6, 10, 14, 18]  
Thus the correct answer would equal to = (12+6+10+14+18) % (10^9+7) = 60