

11.Count of 3 Multiples

Write a program to find the count of 3 multiples in a given input integer array.

Include a function named **divisibleBy3** that accepts 2 arguments and returns an int. The first argument is the input array and the second argument is an int that corresponds to the size of the array. The function returns an int that corresponds to the count of 3 multiples.

If the size of the array is negative or if any element in the array is negative, print “Invalid Input” and terminate the program.

Input and Output Format:

Input consists of n+1 integers. The first integer corresponds to n, the number of elements in the array. The next 'n' integers correspond to the elements in the array.

Output consists of an integer that corresponds to the count of 3 multiples

Assume that the maximum number of elements in the array is 20.

Sample Input 1:

8
1
6
3
5
61
80
102
9

Sample Output 1:

4

Sample Input 2:

-5

Sample Output 2:

Invalid Input

Sample Input 3:

5
23
2
-200

Sample Output 3:

Invalid Input

12.Odd Even Average

The Owner of a block visited the Layout and found that he has some plot numbers of his own and some are odd numbers and some are even numbers. He is maintaining the details in a file in the system. For the password protection our owner has followed one formula. He calculated the sum of his even numbers plot and sum of odd numbers plot and found the average of those two and he used that average as his password for the details file. Find the password that our owner has arrived.

Include a function named **avgOddEvenSum** that accepts 2 arguments and returns a float. The first argument is the input array and the second argument is an int that corresponds to the size of the array. The function returns a float that corresponds to the average of the array.

If the size of the array is negative or if any element in the array is negative , print “Invalid Input” and terminate the program.

Input and Output Format:

Input consists of n+1 integers. The first integer corresponds to n, the number of elements in the array. The next 'n' integers correspond to the elements in the array.

Output consists of a floating point number that corresponds to the average. It is displayed correct to 2 decimal places.

Assume that the maximum size of the array is 20.

Sample Input 1:

5
1
2
3
4
5

Sample Output 1:

7.50

Sample Input 2:

-5

Sample Output 2:

Invalid Input

Sample Input 3:

5
23
2
-5

Sample Output 3:

Invalid Input

13.Decimal Conversion

Write a program to convert a given input binary number to decimal.

Include a function named **convertToDecimal** that accepts an integer argument and returns an integer that corresponds to the decimal representation of the input number. If the input value is not a binary value or if the input is negative or if the input is greater than 11111, the function returns -1.

If the function returns -1, print Invalid Input.

Input and Output Format:

Input consists of a single integer that corresponds to the binary representation of a number.

Output consists of a single integer that corresponds to the decimal equivalent of the given number.

Refer sample output for formatting specifications.

Sample Input 1:

1100

Sample Output 1:

12

Sample Input 2:

101010

Sample Output 2:

Invalid Input

Sample Input 3:

1201

Sample Output 3:

Invalid Input

14.Arithmetic Operation

Write a program to perform a specific arithmetic operation

Include a function named **performArithmeticOperation** that accepts 3 integer arguments and returns an integer that corresponds to the result. The first and second arguments correspond to the input numbers and the third argument corresponds to the choice of arithmetic operation.

If argument 3 =1, calculate the sum of input1 and input2

If argument 3 =2, calculate the difference of input1 and input2

If argument 3 =3, calculate the product of input1 and input2

If argument 3 =4, calculate the quotient of input1 divided by input 2

If the first two argument's values is negative or greater than 32767 , the function returns -1.

If the third argument's value is not in the range 1 to 4, the function returns -1.

If the function returns -1, print Invalid Input.

Input and Output Format:

Input consists of 3 integers.

Output consists of an integer.

Refer sample output for formatting specifications.

Sample Input 1:

4

12

3

Sample Output 1:

48

Sample Input 2:

-67

2

1

15.digitFactorial

Read the question carefully and follow the input and output format.

In a given input number , find out the factorial of each individual digit and assign it to output array.

Input and Output Format:

Input consists of a single integer. Output consists of an Integer array, the individual factorials.

Print "Number too large" when the given input numbers is greater than 32767 .

Print "Number too small" when the given input is a negative number.

Include a function named digitFactorial(int number) whose return type is void.

The output array is stored in a global variable named factorial.

Sample Input 1:

123

Sample Output 1:

1

2
6

Sample Input 2:

-2526

Sample Output 2:

Number too small

16.searchKeys

Read the question carefully and follow the input and output format.

Given an integer array, first index represents the key & second index represents the value. Find keys for the given value.

Input and Output Format:

First line of input consists of n, the number of elements. Next n lines correspond to the array elements. The next line consists of an integer that represents the value to be searched.

Output consists of an integer array.

- 1) Print "Invalid array size" when size of the array is negative and terminate the program.
- 2) Print "Invalid input" when there are any negative numbers available in the input array and terminate the program.
- 3) Print "Key not found" when there are no keys found.

Include a function named searchKeys(int array[], int size) whose return type is void.

The output array is stored in a global variable named found.

Sample Input 1:

8
1
4
2
4
3
4
5
6
4

Sample Output 1:

1
2
3

Sample Input 2:

5
5

6
7
8
9
-5

Sample Output 2:

Key not found

17.passCount

Read the question carefully and follow the input and output format.

Given a input array, First index Represents RollNo second index represents Mark and so on. Write a program to find the number of students who had cleared the exam.

Note : If marks ≥ 70 then He /she Cleared the exam. Array size is always even.

Input and Output Format :

First line of input consists of n, the number of elements. Next n lines correspond to the array elements. Output consist of an integer,

- 1) Print "Invalid array size" when size of the array is a negative number and terminate the program.
- 2) Print "Invalid input" when there is any negative number available in the input array and terminate the program.

Include a function named passCount(int array[], int size) whose return type is an integer , the count.

Sample Input 1:

8
1
70
2
55
3
75
4
80

Sample Output 1:

3

Sample Input 2:

5
6
2
8
-2

Sample Output 2:

Invalid input

18.5 Multiples --- Average

Write a program to find the average of multiples of 5 upto 'n'. n is given as input.

Include a function named **findAverageBy5s** that accepts an integer argument and returns a float that corresponds to the average of multiples of 5.

If the input value is negative or greater than 32767, print Invalid Input and terminate the program.

Input and Output Format:

Input consists of a single integer.

Output consists of a floating point number. Output is displayed correct to 2 decimal places.

Refer sample output for formatting specifications.

Sample Input 1:

10

Sample Output 1:

7.50

Sample Input 2:

-67

Sample Output 2:

Invalid Input

19.Sum of Odd Digits

Write a program to find the sum of the odd digits in a number.

Include a function named **sumOddDigits** that accepts an integer argument and returns an integer that corresponds to the sum of the odd digits. The function returns -1 if the input is less than zero or if it is greater than 32767.

If the function returns -1, print "Invalid Input".

Input and Output Format:

The input consists of an integer.

The output consists of an integer that corresponds to the sum of the odd digits in the number.

Sample Input 1:

3487

Sample Output 1:

10

Sample Input 2:

-8

Sample Output 2:

Invalid Input

20.Largest Array

Write a program which takes two arrays of the same size as a input and compares the first element of first array with the first element of second array and stores the largest of these into the first element of the output array. Repeat the process till the last element of the first array is checked with the last element of the second array.

Include a function named **largestArray** that accepts 3 arguments and its return type is void. The first argument is input array 1, the second argument is input array 2 and the third argument is an int that corresponds to the size of the array. The output array is stored in a global variable named output1.

If the size of the array is negative or if any element in the array is negative, print “Invalid Input” and terminate the program.

Input and Output Format:

Input consists of n+1 integers. The first integer corresponds to n, the number of elements in the array. The next 'n' integers correspond to the elements in the array.

Output consists of the largest array.

Assume that the maximum number of elements in the array is 20.

Sample Input 1:

4
2
1
3
4
1
9
2
8

Sample Output 1:

2
9
3
8

Sample Input 2:

-5

Sample Output 2:

Invalid Input

Sample Input 3:

5
23
2
-200

Sample Output 3:

Invalid Input