

41.University Type

Write a program to find if the student is eligible for first, second or third grade universities by finding the average of their marks given in the input integer array.

Grade should be calculated as given below :

Average >80 First Grade University

Average >60 Second Grade University

Otherwise Third Grade University

Include a function named **calculateGrade** that accepts 2 arguments and returns an integer. 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 integer that corresponds to the university type. The function returns 1 if the student is eligible for First Grade university, returns 2 if the student is eligible for Second Grade University, returns 3 if the student is eligible for Third Grade University and returns -1 if the average is greater than 99.

If the size of the array is negative or if any element in the array is negative or if the average marks scored by the student is greater than 99 , 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 string --- “First Grade University” or “Second Grade University” or “Third Grade University” or “Invalid Input”.

Assume that the maximum size of the array is 20.

Sample Input 1:

5
92
87
78
74
80

Sample Output 1:

First Grade University

Sample Input 2:

-5

Sample Output 2:

Invalid Input

Sample Input 3:

5
23

2
-5

Sample Output 3:

Invalid Input

42.Interchange Array

Write a program to interchange the first element in the array with the last element in the array. Repeat the process till the middle of the array.

Include a function named **interchangeArray** that accepts 2 arguments and its return type is void. The first argument is the input array and the second argument is an int that corresponds to the size 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 the interchanged array.

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

Sample Input 1:

4
2
1
3
4

Sample Output 1:

4
3
1
2

Sample Input 2:

-5

Sample Output 2:

Invalid Input

Sample Input 3:

5

23
2
-200

Sample Output 3:

Invalid Input

43. Armstrong Number

Write a program to find whether the given input number is an Armstrong Number.

Include a function named **checkArmstrong** that accepts an integer and returns an integer. The function returns

1. yes if the input is an Armstrong number
2. no if the input is not an Armstrong number
3. Invalid Input if the input is a negative number or if the input is not a 3-digit number.

Print Invalid Input if the function returns -1.

Input and Output Format:

Input consists of a single integer.

Refer sample output for formatting specifications.

Sample Input 1:

153

Sample Output 1:

yes

Sample Input 2:

161

Sample Output 2:

no

Sample Input 3:

2345

Sample Output 3:

Invalid Input

44. Factorial

Write a program to find the factorial of a given number.

Include a function named **findFactorial** that accepts an integer argument and returns an integer that corresponds to factorial. If the input value is negative or greater than 10, the function returns -1.

If the function returns -1, print Invalid Input.

Input and Output Format:

Input consists of a single integer.

Output consists of an integer.

Refer sample output for formatting specifications.

Sample Input 1:

4

Sample Output 1:

24

Sample Input 2:

-67

Sample Output 2:

Invalid Input

45. Salary Calculation

Jim got his salary. His salary calculations are as follows.

From his Basic amount he gets 50% of his basic for house Rent allowances and 75% of his basic as special allowances . If the number of days he worked is 31 he gets 500 extra. Write a program to calculate his gross salary after calculating all his salary split up.

Include a function named **calculateGross** that accepts 2 integer arguments and returns a float. The first integer corresponds to Jim's basic salary and the second integer corresponds to the number of days Jim has worked. The function returns a float that corresponds to the gross salary.

Print Invalid Input and terminate the program in the following cases:

1. Basic salary is greater than 10000
2. Number of working days is greater than 31
3. Basic salary is negative
4. Number of working days is 0 or negative

Input and Output Format:

Input consists of 2 integers. The first integer corresponds to Jim's basic salary and the second integer corresponds to the number of days he has worked.

Output consists of a single float that corresponds to Jim's gross salary. The gross salary is displayed correct to 2 decimal places.

Sample Input 1:

5000

30

Sample Output 1:

11250 .00

Sample Input 2:

5000

0

Sample Output 2:

Invalid Input

46.Perfect Square

Write a program to find whether the given input number is a perfect square without using sqrt function.

Include a function named **checkPerfectSquare** that accepts an integer and returns an integer. The function returns

1. 1 if the input is a perfect square
2. 0 if the input is not a perfect square
3. -1 if the input is a negative number

Print Invalid Input if the function returns -1.

Input and Output Format:

Input consists of a single integer.

Refer sample output for formatting specifications.

Sample Input 1:

36

Sample Output 1:

yes

Sample Input 2:

40

Sample Output 2:

no

Sample Input 3:

-2345

Sample Output 3:

Invalid Input

47.Find Index

Write a program to find the index of a particular number in a given input array.

Include a function named **findIndex** that accepts 3 arguments and returns an int. The first argument is the input array, the second argument is an int that corresponds to the size of the array and the third argument is the element to be searched for. The function returns the corresponding index if the search element is present in the array and returns -1 if the search element is not present in 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+2 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. The last integer corresponds to the element whose count needs to be found.

Output consists of an integer that corresponds to the index of the search element if it is present. Else, print 'not found'.

Refer sample output for formatting specifications.

Assume that the maximum number of elements in the array is 20 and that all elements in the array are unique.

Sample Input 1:

8
2
1
3
8
6
12
10
19
8

Sample Output 1:

3

Sample Input 2:

8

2

1

3

8

6

12

10

19

80

Sample Output 2:

not found

Sample Input 3:

-5

Sample Output 3:

Invalid Input

Sample Input 4:

5

23

2

-200

Sample Output 4:

Invalid Input

48.Descending Order Sort

Write a program to sort the given array in descending order.

Include a function named **sortArray** that accepts 2 arguments and its return type is void. The first argument is the input array and the second argument is an int that corresponds to the size of the array .

If the size of the array is negative or if any of the elements in the array are 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 array.

Refer sample output for formatting specifications.

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

Sample Input 1:

8
1
6
3
5
8
10
4
9

Sample Output 1:

10
9
8
6
5
4
3
1

Sample Input 2:

-5

Sample Output 2:

Invalid Input

49.endWithThree

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

Given an input array, Find out the count of numbers that ends with 3.

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 the count of numbers that ends with 3.

Print "Invalid array size" when size of the array is a negative number and terminate the program

Print "Invalid input" when there is any negative number available in the input array and terminate the program.

Include a function named `endWithThree(int numbers[], int size)` whose return type is integer.

Sample Input 1:

5
23
353
33
12
14

Sample Output 1:

3

Sample Input 2:

5
1
7
3
-8

Sample Output 2:

Invalid input

50.feeCalculation

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

Student Fees is calculated according to the student's 10th marks. The student will get discount in fees as follows :

Marks discount(%)

>90 - 50%

81-90 - 25%

70-80 - 10%

<70 - 0%

Calculate the fees according to above table.

Note:

Formula : $\text{fees} - (\text{fees} * \text{discount}(\%))$

Include a function named `feeCalculation(int fee,int marks)` that returns an integer that corresponds to the fee to be paid.

Print "Invalid mark" if the mark is greater than 100

Print "Invalid fee" if the fee is greater than 32767

Print "Invalid input" if any of the input is negative

Input and Output Format :

First line of input represents the fee, second line of input represents the marks of student.

Sample Input 1:

10000

95

Sample Output 1:

5000

Sample Input 2:

15896

101

Sample Output 2:

Invalid mark