

CS19003 Programming and Data Structures Laboratory

Assignment 3

-
- ▷ Name the files as `<assignment no>_<question no>_<roll no>.c`, without the `<` and `>`. Consult your mentor for any confusion. Penalty if the file names do not stick to this convention.
 - ▷ This is an assignment on loops/iterations. Restrict yourself to the concepts taught in the theory classes so far. Do not use arrays, functions or any other advanced concept.
-

1. Write a program for a simple calculator which prints the following message on the screen and waits for the user input:

Enter 1 for addition, 2 for subtraction, 3 for multiplication, 4 for division. Enter 5 to quit:

If the user enters an integer between 1 and 4, the program asks for the operands of the corresponding operation, displays the result (or prints an appropriate error message if the operands are illegal), and prints the above prompt. If the user prints 5, the program terminates. If the user enters something other than an integer in the range 1-5, the program prints “invalid input” and prints the above prompt again. The operands are all floating point numbers.

Sample input/output:

Enter 1 for addition, 2 for subtraction, 3 for multiplication, 4 for division. Enter 5 to quit: 2

Enter the first operand: 2

Enter the second operand: -5

difference=7.000000

Enter 1 for addition, 2 for subtraction, 3 for multiplication, 4 for division. Enter 5 to quit: 2

Enter the first operand: -3

Enter the second operand: 4

difference=-7.000000

Enter 1 for addition, 2 for subtraction, 3 for multiplication, 4 for division. Enter 5 to quit: 3

Enter the operands: 3 5.2

product=15.599999

Enter 1 for addition, 2 for subtraction, 3 for multiplication, 4 for division. Enter 5 to quit: 4

Enter the first operand: 3

Enter the second operand: 0

Division by zero error!

Enter 1 for addition, 2 for subtraction, 3 for multiplication, 4 for division. Enter 5 to quit: 4

Enter the first operand: 2

Enter the second operand: 3

division result=0.666667

Enter 1 for addition, 2 for subtraction, 3 for multiplication, 4 for division. Enter 5 to quit: 6

Invalid input!

Enter 1 for addition, 2 for subtraction, 3 for multiplication, 4 for division. Enter 5 to quit: 5

[10 points]

2. Write a program to compute the prime factorization of a positive integer that is entered by the user. The program should print one prime factor along with its multiplicity in a fresh line.

Sample input/output:

Enter a positive integer: 9000

Prime factor: 2, multiplicity: 3

Prime factor: 3, multiplicity: 2

Prime factor: 5, multiplicity: 3

[10 points]

3. Write a program with the following input output behavior. It first asks the user for an integer n. After the user enters n, the program takes in the co-ordinates of n points on the two-dimensional real plane through the keyboard. Assume that each co-ordinate is an integer in the range [-1000, 1000]. The program then prints the co-ordinates of the bottom-left and the top-right vertices of the smallest axis-parallel rectangle (i.e., rectangle whose sides are parallel to the X and Y axes) that contains all the n points. We will say that a point on a side of a rectangle is contained by the rectangle.

Sample input/output:

Enter number of points: 5

Enter x-coordinate of point 1: -1

Enter y-coordinate of point 1: 2

Enter x-coordinate of point 2: 3

Enter y-coordinate of point 2: -4

Enter x-coordinate of point 3: 2

Enter y-coordinate of point 3: 3

Enter x-coordinate of point 4: 0

Enter y-coordinate of point 4: 1

Enter x-coordinate of point 5: -4

Enter y-coordinate of point 5: 4

Rectangle: (-4,-4), (3,4)

[10 points]