Programming Fundamental

Assignment - 05

Instructions:

- It is individual assignment, so try to do it by yourself.
- Assignment should be in zip file named as RollNumber_Name_PF.zip and name of your file should be according to tasks given in this assignment.
- Submit your assignment on google classroom before deadline **Wednesday 05 October 2022, 11:59pm**. Late submissions will not be considered.
- Feel free to ask any question.
- Total marks of assignment are 100, 10 each.

Task# 01:

Assume that you want to generate a table of multiples of any given number. Write a program that allows the user to enter the number and then generates the table, formatting it into 10 columns and 20 lines. Interaction with the program should look like this (only the first three lines are shown):

```
Enter a number: 7
    7   14   21   28   35   42   49   56   63   70
   77   84   91   98  105  112  119  126  133  140
  147  154  161  168  175  182  189  196  203  210
```

Task# 02:

Suppose you give a dinner party for six guests, but your table seats only four. In how many ways can four of the six guests arrange themselves at the table?

Any of the six guests can sit in the first chair. Any of the remaining five can sit in the second chair. Any of the remaining four can sit in the third chair, and any of the remaining three can sit in the fourth chair. (The last two will have to stand.)

So the number of possible arrangements of six guests in four chairs is 6*5*4*3, which is 360.

Write a program that calculates the number of possible arrangements for any number of guests and any number of chairs. (Assume there will never be fewer guests than chairs.) Don't let this get too complicated. A simple loop should do it.

Task# 03:

Create a four-function calculator for fractions. Here are the formulas for the four arithmetic operations applied to fractions:

Addition:	$a/b + c/d = (a*d + b*c) / (b*d)$
Subtraction:	$a/b - c/d = (a*d - b*c) / (b*d)$
Multiplication:	$a/b * c/d = (a*c) / (b*d)$
Division:	$a/b / c/d = (a*d) / (b*c)$

The user should type the first fraction, an operator, and a second fraction. The program should then display the result and ask whether the user wants to continue.

Do you wish to continue (y/n)?

y mean executes the above program again. n mean end the program.

Task# 04:

Write a program that asks the user to enter today's sales for five stores. The program should then display a bar graph comparing each store's sales. Create each bar in the bar graph by displaying a row of asterisks. Each asterisk should represent $100 of sales. Here is an example of the program's output.

```
Enter today's sales for store 1: 1000 [Enter]
Enter today's sales for store 2: 1200 [Enter]
Enter today's sales for store 3: 1800 [Enter]
Enter today's sales for store 4: 800 [Enter]
Enter today's sales for store 5: 1900 [Enter]

SALES BAR CHART
(Each * = $100)
Store 1: **********
Store 2: ************
Store 3: ******************
Store 4: ********
Store 5: *******************
```

Task# 05:

Write a program with a loop that lets the user enter a series of integers. The user should enter −99 to signal the end of the series. After all the numbers have been entered, the program should display the largest and smallest numbers entered.

Task# 06:

A teacher has asked all his students to line up single file according to their first name.
For example, in one class Amy will be at the front of the line and Yolanda will be at the end.

Write a program that prompts the user to enter the number of students in the class, then loops to read that many names. Once all the names have been read it reports which student would be at the front of the line and which one would be at the end of the line. You may assume that no two students have the same name.
Input Validation: Do not accept a number less than 1 or greater than 25 for the number of students.

Task# 07:

Write a program that asks the user to enter the amount that he or she has budgeted for a month. A loop should then prompt the user to enter each of his or her expenses for the month and keep a running total. When the loop finishes, the program should display the amount that the user is over or under budget.

Task# 08:

Write a program that displays a weekly payroll report. A loop in the program should ask the user for the employee number, gross pay, state tax, federal tax, and FICA with-holdings.
The loop will terminate when 0 is entered for the employee number.
After the data is entered, the program should display totals for gross pay, state tax, federal tax, FICA withholdings, and net pay.

Input Validation: Do not accept negative numbers for any of the items entered. Do not accept values for state, federal, or FICA withholdings that are greater than the gross pay. If the sum state tax + federal tax + FICA withholdings for any employee is greater than gross pay, print an error message and ask the user to reenter the data for that employee.

Task# 09:

A prime number is a number that is only evenly divisible by itself and 1. For example, the number 5 is prime because it can only be evenly divided by 1 and 5. The number 6, however, is not prime because it can be divided evenly by 1, 2, 3, and 6.
Write a program, which takes an integer from user and print "Number is Prime" if the number is a prime number, or "Number is not Prime" otherwise.

Task#10:

Write a C++ program that reads an integer between 0 and 65535 from the keyboard and uses it to seed a random number generator. Then output 20 random numbers between 1 and 100 on screen.