Programming Fundamentals *COURSE & LAB* – Spring 2022

(BS-IT-F21 Morning)

Assignment # 2

Assigned on: ***Tuesday, September 27, 2022***

Submission Deadline of assignment : ***Tuesday, October 04, 2022 (till 11:59 PM)***

Instructions:

- **This is an individual assignment. Absolutely NO collaboration is allowed. Any traces of plagiarism/cheating would result in an "F" grade in this course.**
- Do **NOT** copy even a single line of code from any other person or book or Internet or any other source.
- needs to be submitted in **Soft** form. The **Submission Procedure** is given on last page.
- Clearly mention your **Name**, **Roll Number** and **Section** in comments at the **top** of **each CPP file**.

Question # 1

Write a C++ program which prints a hollow rectangle after taking its *height* and *width* from the user. For example, if the user specifies height to be **6** and width to be **10**, the following output should be produced:

```
**********
*        *
*        *
*        *
*        *
**********
```

In order to display the above pattern, your program should use ONLY following THREE cout statements:

```
cout << "*";     cout << " ";     cout << endl;
```

Input validation: Your program should make sure that both height and width are ***even numbers****. Moreover, the value of* ***height should be at least 4*** *and value of* ***width should be at least 6****. In case of invalid input, your program should keep prompting the user again and again till the user provides valid input.*

Question # 2

Write a C++ program which takes a positive integer *N* from the user, and prints the multiplication table of size *N* by *N*. For example, if the user enters **10**, the following table should be displayed by your program.

```
     1   2   3   4   5   6   7   8   9  10
   +-----------------------------------------
 1 |   1   2   3   4   5   6   7   8   9  10
 2 |   2   4   6   8  10  12  14  16  18  20
 3 |   3   6   9  12  15  18  21  24  27  30
 4 |   4   8  12  16  20  24  28  32  36  40
 5 |   5  10  15  20  25  30  35  40  45  50
 6 |   6  12  18  24  30  36  42  48  54  60
 7 |   7  14  21  28  35  42  49  56  63  70
 8 |   8  16  24  32  40  48  56  64  72  80
 9 |   9  18  27  36  45  54  63  72  81  90
10 |  10  20  30  40  50  60  70  80  90 100
```

Note: The output of your program should follow the above output format exactly. You will have to use appropriate ***stream manipulator(s)*** *to achieve this.*

Question # 3

Write a C++ program which prints the following window pattern after taking *height* and *width* from the user. For example, if the user enters height to be **5** and width to be **11**, the following pattern should be displayed on screen:

```
++++++++++
+ + + + + +
++++++++++
+ + + + + +
++++++++++
```

In order to display the above pattern, your program should use ONLY following THREE cout statements:

```
cout << "+";     cout << " ";     cout << endl;
```

Input validation: Your program should make sure that both height and width are ***odd numbers****. Moreover, the value of* ***height should be at least 3*** *and value of* ***width should be at least 5****. In case of invalid input, your program should keep prompting the user again and again till the user provides valid input.*

Question # 4

Write a C++ program which prints the following square pattern by taking the size (dimensions) from the user. For example, if the user enters the size to be **6**, the following 6 by 6 pattern should be displayed on screen:

```
111111
011111
001111
000111
000011
000001
```

In order to display the above pattern, your program should use ONLY following THREE cout statements:

```
cout << "0";     cout << "1";     cout << endl;
```

Input validation: *Your program should make sure that the size is an **even number greater than or equal to 2**. In case of invalid input, your program should keep prompting the user again and again till the user provides valid input.*

Question # 5

A right triangle can have sides that are all integers. A set of three integer values for the sides of a right triangle is called a *Pythagorean triple*. These three sides must satisfy the relationship that the sum of the squares of two of the sides is equal to the square of the hypotenuse. Write a C++ program to find and display all Pythagorean triples for side1, side2 and hypotenuse all no larger than 200. (*Hint: Use a **triple-nested for loop** that checks all possibilities.*)

Question # 6

3. Winning Division

Write a program that determines which of a company's four divisions (1, 2,3, and 4) had the greatest sales for a quarter. It should include the following two functions, which are called by `main`:

- `double getSales()` is passed the no. of a division. It asks the user for a division's quarterly sales figure, validates the input, then returns it. It should be called once for each division.
- `void findHighest()` is passed the four sales totals. It determines which is the largest and prints the no. of the high-grossing division, along with its sales figure.

Input Validation: Do not accept dollar amounts less than $0.00.

Question # 4

10. **Future Value**

Suppose you have a certain amount of money in a savings account that earns compound monthly interest, and you want to calculate the amount that you will have after a specific number of months. The formula, which is known as the future value formula, is

$$F = P \times (1 + i)^t$$

The terms in the formula are as follows:

- F is the **future value** of the account after the specified time period.
- P is the **present value** of the account.
- i is the **monthly interest rate.**
- t is the **number of months.**

Write a program that prompts the user to enter the account's present value, monthly interest rate, and the number of months that the money will be left in the account. The program should pass these values to a function named `futureValue` that returns the future value of the account, after the specified number of months. The program should display the account's future value.

Question # 7

13. **Days Out**

Write a program that calculates the average number of days a company's employees are absent. The program should have the following functions:

- A function called by `main` that asks the user for the number of employees in the company. This value should be returned as an `int`. (The function accepts no arguments.)
- A function called by `main` that accepts one argument: the number of employees in the company. The function should ask the user to enter the number of days each employee missed during the past year. The total of these days should be returned as an `int`.
- A function called by `main` that takes two arguments: the number of employees in the company and the total number of days absent for all employees during the year. The function should return, as a `double`, the average number of days absent. (This function does not perform screen output and does not ask the user for input.)

Input Validation: Do not accept a number less than 1 for the number of employees. Do not accept a negative number for the days any employee missed.

Question # 8

Write a C++ program that uses the following **FOUR** functions to calculate and display the area of a rectangle whose length and width are taken from the user. Each of these functions is explained below:

- **`int isValid (double num, double start, double end);`**
  This function will return **true** if the value of the variable **`num`** is in the range from **`start`** to **`end`** (both inclusive), and it should return **false** otherwise. There should be NO input or output performed within this function.

- **`int getRadius ();`**
  This function will prompt the user to enter the **radius** of a circle in the range from **10.0** to **50.0** (both inclusive). In case of invalid input, this function should keep prompting the user again and again till the user provides valid input. This function MUST use the function **`isValid(…)`** (which you have implemented above) to validate the input. The valid input provided by the user should be returned from this function.

- **`void calculateArea (int radius, double *area);`**
  This function will receive the radius of the circle (see the first parameter). It should calculate and store the area of the circle into the *reference/pointer parameter* **`area`** (see the second parameter). There should be NO input or output performed within this function.

- **`void displayArea (int area);`**
  This function will display the area of the circle on the screen. Use meaningful label(s) in output.

These *good programming practices* will also have their (significant) weightage in the marking of this assignment:

- Comment your code intelligently. **Uncommented code will NOT be given any credit.**
- Indent your code properly.
- Use meaningful variable and function names. Use **camelCase** notation.
- Use meaningful prompt lines/labels for input/output.

☺ GOOD LUCK! ☺