NATIONAL UNIVERSITY OF COMPUTER & EMERGING SCIENCESISLAMABAD CAMPUS
Programming Fundamentals - FALL 2023

ASSIGNMENT- 3

Due Date: 31-10-2023 (11:59 PM)

Please follow the following submission instructions. Failure to submit according to the format would result in a deduction of marks. **Submissions other than Google classroom (e.g., email etc.) will not be accepted**. **No late submission will be accepted. Correct and timely submission of the assignment is the responsibility of every student; hence no relaxation will be given to anyone.**

Instructions: **Total Marks: 145**

1. Assignments are to be done individually. You must complete this assignment by yourself. You cannot work with anyone else in the class or with someone outside of the class. The code you write must be your own and you must have an understanding of each part you code. You are encouraged to get help from the instructional staff through email, on google classroom or individually visiting their respective offices.
2. The primary focus of this assignment is to give you practice with different types of loops in C++.
3. Use appropriate data types, operations, and conditional structures for each problem. You cannot use advance constructs like **Arrays / Functions** for this assignment. Built-in Functions are also not allowed.
4. No late assignments will be accepted.
5. Displayed output should be well mannered and well presented. Use appropriate comments and indentationin your source code.

Honor Policy

This assignment is a learning opportunity that will be evaluated based on your ability. Plagiarism cases will be dealt with strictly. Plagiarism of any kind (copying from others, copying from the internet etc.) is **not** allowed. If found plagiarized, both the involved parties will be awarded zero marks in this assignment, all the remaining assignments, or even an F grade in the course.

Submission Guidelines

1. Dear students, we will be using auto-grading tools, so failure to submit according to the below format would result in zero marks in the relevant evaluation instrument.
2. For each question in your assignment, make a separate .cpp file e.g., for question

1, make q1.cpp and so on. Each file that you submit must contain your name, student-id, and assignment # on top of the file in the comments.

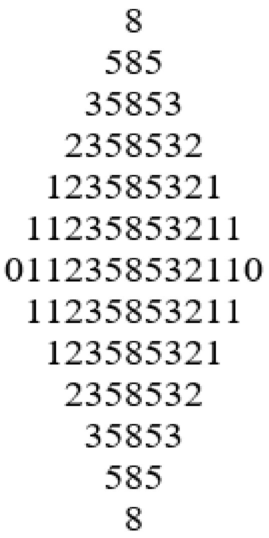
3. Combine all your work in one folder. The folder must contain only .cpp files (no binaries, no exe files etc.).
4. Rename the folder as ROLL-NUM_PROGRAM_SECTION (e.g., 23i-0001_AI/DS_B) and compress the folder as azip file. (e.g., 23i-0001_AI_B.zip).
5. Submit the .zip file on Google Classroom within the deadline.
6. The student is solely responsible to check the final zip files for issues like corrupt file, virus in the file, mistakenly exe sent. If we cannot download the file from Google classroom due to any reason it will lead to zero marks in the assignment.

Note: Start early so that you can finish it on time.

Question No 1 (10 + 5 Marks)

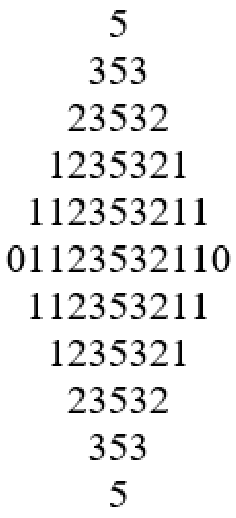
Create a program in C++ that prompts the user to enter a number, verifies if it belongs to the Fibonacci sequence, and then generates a diamond pattern based on the Fibonacci series.
Procedure:
Step 1: Request the user to input a number to be checked.
Step 2: Determine whether the input number is part of the Fibonacci sequence.
Step 3: Accumulate all Fibonacci numbers up to the user-provided number into a single integer variable.
Step 4: Utilize this number and the modulo operator to display a diamond pattern.

Example **:**
Input : 8
Number : 0112358
Pattern:



```
      8
     585
    35853
   2358532
  123585321
 11235853211
0112358532110
 11235853211
  123585321
   2358532
    35853
     585
      8
```

Input : 5
Number : 011235
Pattern :



```
     5
    353
   23532
  1235321
 112353211
01123532110
 112353211
  1235321
   23532
    353
     5
```

Input : 6
Invalid Input Number is not Fibonacci.
Note : DO IT USING WHILE LOOP ONLY

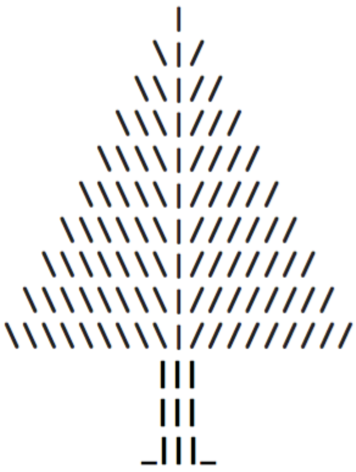
Bonus : If your code is able to print for (13) and (21) input you will get 5 numbers as bonus.

Think about it :)

Question no 2 (10 Marks)

Write a program in C++ to print the following pattern. Make sure your code should be generic and print the following pattern using **FOR LOOP.**

Input of 9 will print the following one.



```
         |
        \|/
       \\|//
      \\\|///
     \\\\|////
    \\\\\|/////
   \\\\\\|//////
  \\\\\\\|///////
 \\\\\\\\|////////
\\\\\\\\\|/////////
        |||
        |||
       _|||_
```

Question no 3: (25 Marks)

Create a simplified C++ program for a two-player Tic Tac Toe game using **Loops**. The game's functionality includes:

1. Displaying a 3x3 game board with a numbered grid to help players select their moves.
2. Allowing two players to take turns, one using 'X' and the other 'O.'
3. Verifying the validity of each move (i.e., the chosen cell is within the board boundaries).
4. Checking for a win condition or a draw after each move.
5. Prompting players to make their moves by specifying the 0-9.
6. Displaying the updated game board after each move.
7. Announcing the winner or a draw when the game ends.
8. Ending the game automatically after nine moves if no winner is determined.
9. Providing a player-friendly experience with clear prompts and messages.

Your task is to code and add the necessary logic to determine the game's outcome (win, draw, or continue) after each move. Also, it should also print the board after every move. Ensure that the game runs smoothly and accurately. The game should continue until a player wins or the game ends in a draw.
Note: Arrays or Functions are not allowed

Question no 4: (10 Marks)

Background:

The Josephus Problem is a classic mathematical puzzle with various applications in computer science and beyond. It involves a group of people standing in a circle, and one person is eliminated in a sequential manner until only one remains. The challenge is to determine the position of the last person standing, given the total number of people and a fixed counting interval.

Scenario:
Imagine you are a computer science student working on a programming project. Your professor assigns you the task of solving the Josephus Problem using a programming language of your choice. The professor wants to assess your problem-solving skills and understanding of iterative algorithms.

Objectives:

1. Write a program in C++ to find the position of the last person standing in the Josephus Problem.
2. Implement the program using an iterative approach.

3. Handle user input for the number of people and the counting interval.
4. Ensure that the program produces the correct result and runs efficiently for reasonably large inputs.

Outputs:

```
Enter the number of people (n): 10
Enter the step (k): 2
The last person is standing at position: 5

Enter the number of people (n): 120
Enter the step (k): 5
The last person is standing at position: 33
```

Note : Make it use WHILE LOOP only.

Question no 5: (10 Marks)

Write a C++ program that generates a prime spiral of numbers for a given positive integer N. The prime spiral should start from 1 and go up to N^2, and for each number in the spiral position, the function should print "P" if it's prime or "." if it's not.

For example

When N is 4, the generated prime spiral should look like this:

P	P	P	.
P	.	P	.
.	.	P	.
P	.	.	.

When N is 6, the generated prime spiral should look like this:

P	P	P	.	P	.
P	.	.	.	P	.
P	.	.	.	P	.
P	.	.	.	P	.
.	.	.	.	P	.
P	.	.	.	.	.

Note : Arrays or Functions are not allowed.

Question no 6: (15 Marks)

Maclaurin Series for sine function:

The Maclaurin series is a specific case of the Taylor series where it is centered at x = 0. For the sine function, the Maclaurin series is as follows:

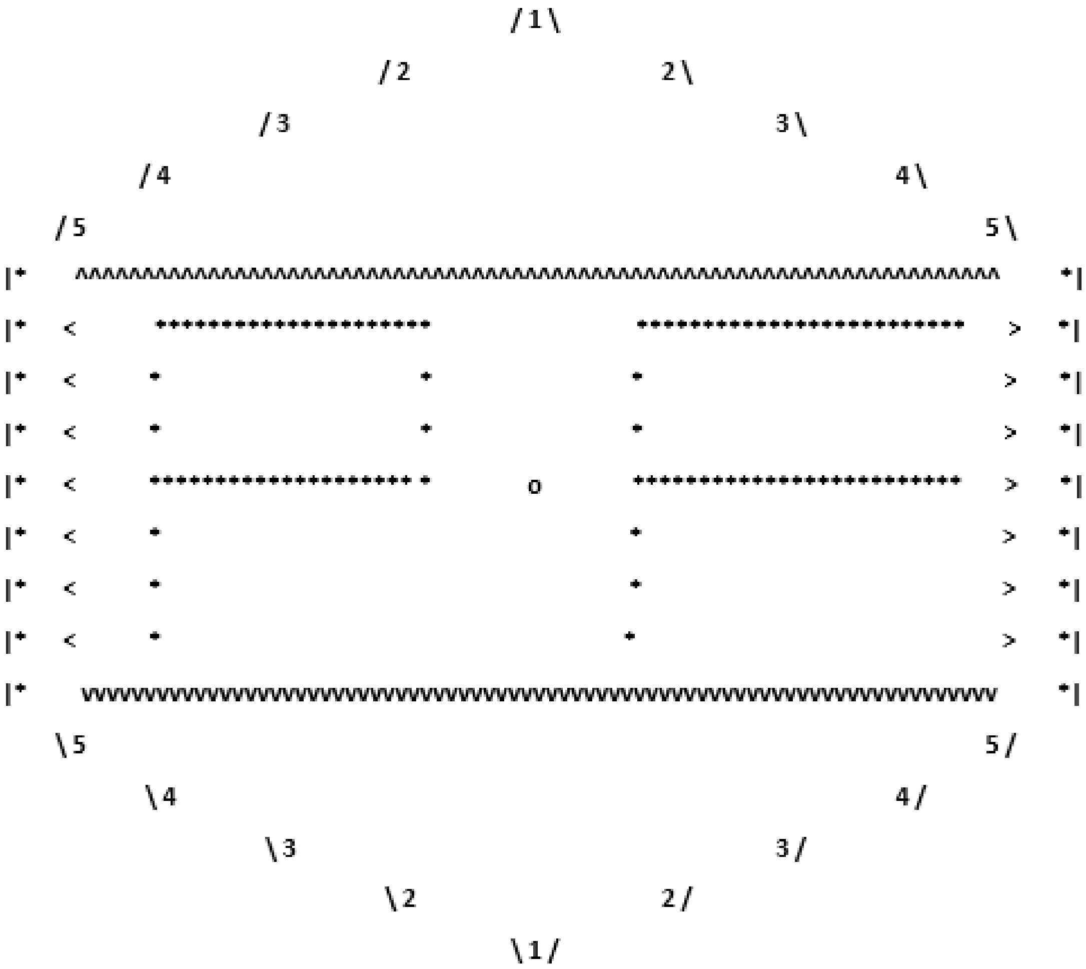
sin(x) = x - (x^3 / 3!) + (x^5 / 5!) - (x^7 / 7!) + ...

Write a program in C++ that should take user input for the value of 'x' and the number of terms 'n' to use in the Maclaurin series, and then calculate and display the approximation of sin(x) based on the Maclaurin series formula.

Note : No built-in functions are allowed.

Question No 7: (15 Marks)

Write a program in C++ to print the following pattern.



```
                                           /1\
                                /2                     2\
                     /3                                          3\
           /4                                                              4\
    /5                                                                             5\
|*    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^     *|
|*   <       **********************                  ***************************     >   *|
|*   <       *                     *                 *                               >   *|
|*   <       *                     *                 *                               >   *|
|*   <      ********************* *         o       ***************************      >   *|
|*   <       *                                       *                               >   *|
|*   <       *                                       *                               >   *|
|*   <       *                                      *                                >   *|
|*    vvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv     *|
    \5                                                                             5/
            \4                                                             4/
                      \3                                         3/
                                \2                     2/
                                           \1/
```

Question No 8: (15 Marks)

Imagine you are tasked with developing a base conversion program in C++ that can be used to convert a decimal number into a user-specified base, accommodating both numeric and alphanumeric representations as needed.

Requirements:

1. The program should prompt the user for a decimal number (base 10).
2. The program should prompt the user for the target base, which must be between 2 and 16 (inclusive).
3. The program should use a while loop to perform the conversion.
4. If the target base is between **2 and 10**, the program should output the result as an integer.
5. If the target base is **greater than 10**, the program should output the result as a string using **'A' for 10, 'B' for 11,** and so on, up to **'F' for 15**.
6. The program should validate user inputs and provide error messages for invalid input.

Question no 9: (20 Marks)

You are tasked with creating a C++ program that generates a calendar for a given month and year. The program should display the days of the week and the corresponding dates in a grid format. You need to account for leap years and ensure that different months are handled correctly.

Your program should perform the following tasks:

1. Prompt the user for input:

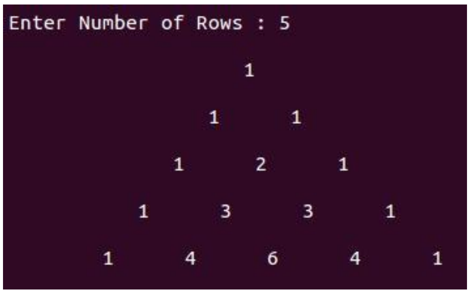
   a) The year (e.g., 2023).
   b) The month (e.g., 4 for April).
   c) Check if the entered year is a leap year and validate that the month entered is in the valid range (1-12). You can use standard leap year rules to determine if a year is a leap year (i.e., divisible by 4 but not divisible by 100, unless divisible by 400).

2. Generate and display the calendar for the specified month and year. The calendar should be displayed in a grid format with the following features:
3. The month and year should be displayed at the top center of the calendar.
4. The days of the week (Sunday, Monday, Tuesday, etc.) should be displayed above the calendar.
5. Each date in the grid should be labeled with the day of the month, and empty spaces should be left for days that fall outside the specified month.
6. Make sure that the dates align correctly with the days of the week.
7. The program should continue running until the user chooses to exit.

Note: Use IOMANIP library and LOOP for making this.

Question No 10: (15 Marks)

Write a program in C++ to generate Pascal's Triangle of a specified number of rows. After calculating the sum and product of all numbers in triangle.



```
Enter Number of Rows : 5
                    1

                 1      1

              1      2      1

           1      3      3      1

        1      4      6      4      1
```

<<GOOD LUCK 😊>>