

**Term Project (30-12-2019)****Submission Date: (09-01-2020)**

**Note 1:** Choose any one project (Individual Project) and email to (morning students [bitf16m543@pucit.edu.pk](mailto:bitf16m543@pucit.edu.pk) & afternoon students [BCSF16A532@pucit.edu.pk](mailto:BCSF16A532@pucit.edu.pk)) by tomorrow 31-12-2019 before 12 PM. We may change your project on any basis, so hurry up to finalize your project

**Note 2:** Students shown interest in C# should implement project in C#

**Note 3:** Students having marks between 50-60 in midterm are not required to do any project

**Project 1: Game of Shogi/ Chess**

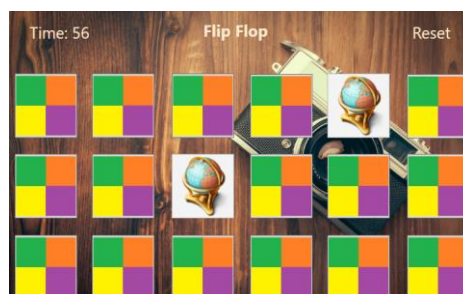
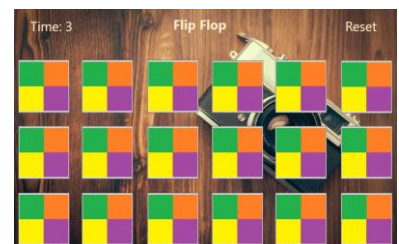
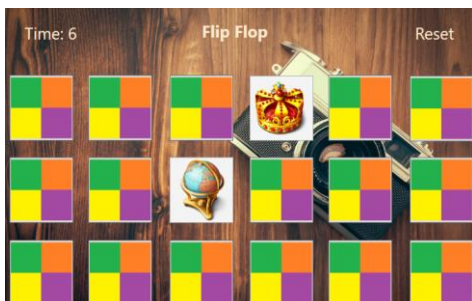
You guys have to make game of Shogi(<https://en.wikipedia.org/wiki/Shogi>) or Chess. Shogi is basically Japanese version of Chess, it will be a challenging task for every one of you, you have to implement OOP in this program using C++. You can either make a piece in this game or represent it using some character.

**Project 2: Flip Flop Game**

You are required to implement the **Flip Flop** game. You can visit this link to get a better understanding of this game. [http://www.primarygames.com/puzzles/match\\_up/flipflopmatch/](http://www.primarygames.com/puzzles/match_up/flipflopmatch/). These screenshots are just for sample, you have to create this game with versatile and high dimensional graphical designs with Windows Form.

Flip Flop game is a picture matching game. Every time the player selects box, a picture is hidden behind that box which is shown or we can simply say that the picture is flipped, that picture remains there until player selects a second box, if the second and first picture are the same, those two pictures remain there (you can even make them vanish but after a delay of few seconds ☹) otherwise these picture are reverted/flipped again with a delay of few seconds ☹. The player wins when no unmatched picture is left. The pictures that you will put on the buttons/boxes should shuffle every time you start a new game. Moreover, a timer should also be there which increments automatically and shows a proper time ☹. When player completes the game, the time, in which he/she completed the game should be shown with score of the player (Score depends upon the time in which he/she completed the game ☹). You have to implement different data members and member functions using OOP concepts. e.g. you can make member functions like move (To take input from user), match Pictures (To match the two pictures selected), play (To start and continue the game), print (to print the game), reset (To reset the game, it will also reset the timer and shuffle the pictures) etc ☹

You can also implement multiple more member functions. More the member functions, more the numbers you will get ☹.



## Project 3: Four in a Line Game

You are required to implement the **Four in a Line** game. You can visit this link to get a better understanding of this game. <https://www.mathsisfun.com/games/connect4.html>

The screenshot is just for sample, you have to create this game with versatile and high dimensional graphical designs with Windows Form. Do everything you can to make your game more attractive ☺.

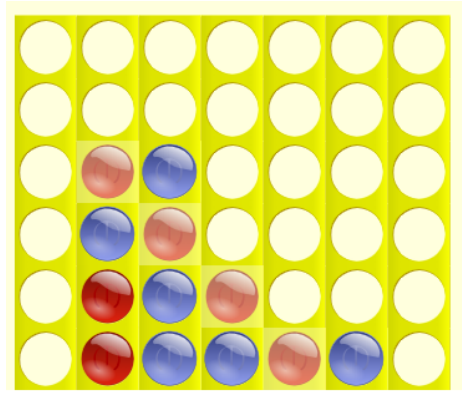
### Introduction:

Four in a line game is played in a **7 X 7** grid. Every time a player selects a column or cell, the cells in that column start to fill from bottom to up. If all the cells in that particular column are filled, then user cannot select that column. The player who succeeds in placing four respective marks horizontally, vertically or diagonally wins the game.

In the following screenshot game is won by the first player denoted by red wins. This game will consist of two stages:

### Stage 1:

Create a class **FourInALine** that will enable you to write a complete program to play the game. The class contains a grid of size **7x7**. The constructor should initialize the empty board. Allow two human players to play the game. Wherever the first player selects a column or cell, place a red ball or any unique thing ☺ in the last unfilled cell of that column. Do similar when second player moves. After each move, determine whether the game has been won or is a draw.



You have to implement different data members and member functions using OOP concepts. e.g. you can make member functions like move (To take input from user), checkDiagonal (To match the cells diagonally and see if a player has won), checkVertical (To match the cells vertically and see if a player has won), checkHorizontal (To match the cells horizontally and see if a player has won), play (To start and continue the game), print (to print the board) etc. You can also implement multiple more member functions. More the member functions, more the numbers you will get.

### Stage 2:

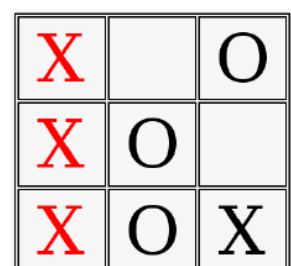
In the first stage of your project you have developed a game Four-In-A-Line for two human players. Now you are required to add a new feature through which a player can play with computer. There should be three modes in this stage (Easy, Medium and Hard). e.g. In Hard mode, program your code in such a way that the computer should give humans a very tough time ☺. Different member functions should be present for different modes and different features. Give option to the user in the start of game whether he wants to play against the computer or not. Select the cell whenever human or computer makes a move respectively.

## Project 4: Tic Tac Toe

You are required to implement the **Tic Tac Toe** game. The screenshot is just for sample. You have to create more versatile and high dimensional graphical designs with Windows Form. Do everything you can to make your game more attractive ☺.

### Introduction:

Tic-tac-toe, also called noughts and crosses, hugs and kisses, zero and kata and many other names, is a pencil-and-paper game for two players, **O** and **X**, who take turns marking the spaces in an **N X N** grid. The player who succeeds in placing three respective marks horizontally, vertically or diagonally wins the game. In the following screenshot game is won by Player X. This game will consist of three stages:



**Stage 1:**

Create a class **TicTacToe** that will enable you to write a complete program to play the game of tic-tac-toe. The class contains a grid of size **N – by – N (e.g. 3X3, 4X4 or 5X5 which is going to be taken as an input from the user)** of integers. The constructor should initialize the empty board. Allow two human players to play the game. Wherever the first player moves, place an **X** in the specified square. Place a **O** wherever the second player moves. Each move must be to an empty square. After each move, determine whether the game has been won or is a draw. You have to implement different data members and member functions using OOP concepts. Eg: you can make member functions like move (To take input from user), checkDiagonal (To match the cells diagonally and see if a player has won), checkVertical (To match the cells vertically and see if a player has won), checkHorizontal (To match the cells horizontally and see if a player has won), play (To start and continue the game), print (to print the board) etc. More the member functions, more the numbers you will get.

**Stage 2:**

In the first stage of your project you developed a game Tic – Tac – Toe for two human players. Now you are required to add a new feature through which a player can play with computer. There should be three modes in this stage (Easy, Medium and Hard). e.g: In Hard mode, program your code in such a way that the computer should give humans a very tough time ☹. Place **X** and **O** whenever human and computer make a move respectively. Different member functions should be present for different modes and different features. Give proper option to the user whether he wants to play against the computer or not.

**Stage 3:**

In the previous two stages of your project you developed the game Tic – Tac – Toe for two human players and for the single player. Now you are required to add a new feature through which a player can save the state of the game whenever he/she wants to do that and resume it from where he/she left the game at any instance of time.

The point to remember is that, a player can save the game many times; for example, the user can save it at time 1 followed by time 2 and so on. And when the user wants to play the game again you have to show him all the saved sequences from one of which will be selected by the user and game resume from that particular state. Additionally, you should provide a facility to remove any of the already existing game state by giving option to the user. A player can save the game at any time of the game, e.g: you can provide the player with a button to save the state of the game. New member functions should be implemented to save, load and delete a game.

Give proper options to the user in the start of the game whether he wants to play a saved game, wants to delete a saved game or wants to play a new game. If he/she selects a new game, then ask him if he wants to play against the computer or not. After that input the grid size from user and then start the game ☹. These features will be implemented by file handling.

**Project 5: CMS**

You are required to implement the **simple and basic CMS** with proper use of file handling. You have been taught operator overloading, so overload all operators that may be used in the class. All class data members should be private and all the data should be stored in files. You can create versatile and high dimensional graphical designs with Windows Form. You have to implement different data members and member functions using OOP concepts. You can also implement multiple more data members and member functions. More the member functions, more the numbers you will get.

It should maintain records of:

- Students
- Teachers

- Classes

Classes to implement:

**1. Student:**

- Name
- ID
- CGPA

**2. Teacher:**

- Name
- ID
- Scale (integer type)

**3. Class:**

- Students (Strength should be variable but use of memory should be optimised.)
- ID
- Teacher

Now write getters, setters, constructors and destructors for each.

**Then your program should display follow menu:**

- 1: Create new class
- 2: Swap Teachers
- 3: Display all CRs
- 4: Add 2 classes
- 5: Add Student to a class
- 6: Delete Student
- 7: Display Class
- 8: Show the better class
- 9: Show Student information
- 10: Show Teacher information

Here is the explanation to each.

**1. Create Class:**

When you enter 1, you should be asked for class id, class teacher Information, no. of students and their Information.

After giving all these information, class should be created.

**2. Swap Teachers:**

When you enter 2, program should display all class ids and ask user to enter two of them so that the teachers of both classes may be swapped with each other.

**Must note that don't use getter to access teacher names. Look for some other way. And yes you can use setter to assign names to classes.**

**3. Show All CRs:**

When you enter 3, your program should display all CRs but be careful while writing this function that data of class may not be changed by this function.

#### **4. Add 2 Classes:**

When you enter 4, program should display all class ids and ask user to enter 2 of those ids and then add these two classes to generate a new class. The teacher for this new class will be the one with greater scale. And Students with highest CGPAs will be added in this class.

#### **5. Add Student to a Class:**

When you enter 5, your program displays all available classes. Then it should ask user to enter the class id of that class in which you want to add new student. Handle all possible wrong entries.

#### **6. Delete Student:**

When you enter 6, the program should ask user to enter student id whom you want to delete. When user enters the id, program first display all information of student and then delete this.

#### **7. Display Class:**

When you enter 7, program should ask class id, and then display all information of the class whose id is entered.

#### **8. Show the Best Class:**

When you enter 8, program should display all available classes and then ask user for 2 classes and then program will display the best class among these. The criteria for better class is, class with more no. of students having CGPA higher than 3.00 is better among these. Program should handle unusual case if there is.

**Hint:** Operator overloading

#### **9. Show Student Information:**

When you enter 9, program should ask for Student id, after that it should display all information of student whose id is entered (name, CGPA, his class id and his teacher name).

#### **10. Show Teacher Information:**

When you enter 10, program should ask for Teacher id, after that it should display all information of Teacher whose id is entered (name, class, his class id and student names in his class).

## **Project 6: Objective Type Paper Generation**

You are required to create a software to generate objective type paper automatically. There are three types of possible questions:

- MCQ [Question contains description, four options and correct option as answer]
- Fill in the blanks [Question contains description including blank somewhere from start to end, a single word correct option as answer, you may use \_ to handle multiple words as single word]
- True/ False [Question contains description only; the answer contains true/ false]

Create these types as child classes of Question Type.

Write three corresponding classes to format questions, again take Question Format as parent class.

Lastly make a class Exam, which may contain any number of questions from three categories. User can select all these parameters at run time:

- # of total questions
- # of MCQs questions
- # of Fill in the blanks questions

- # of True/ false questions

Take three text files having 30+ questions of each type. At run time generate paper and store use in another text file "paper.txt" and store answers in another text file "key.txt".

Use proper header to store information about paper like total questions, and count of each category.