**Lab sheet 12 - Programs using Classes**

**Remember to design your programs with pseudo-code and/or flow diagrams first. Name the programs appropriately.**

# **It is absolutely prohibited to copy anyone else’s design or code. You can ask for help with a particular problem from friends, colleagues, the lecturers in the lab etc but you must write your own design and code the fix to the problem yourself.**

**If you get help from someone or take design/code from the web or elsewhere, you have to comment in your code to state what help you got and from where.**

**Saving Your Projects**

You can create a new project for each program or create them as separate files within the one project.

You should save the programs to a folder called **labsheet 12** within **Programming\Labsheets** folder on your **M drive**.

Every program should have a **comment** at the top stating your name, student ID number, date created, approx number of hours worked on, overall brief description of the program and any known bugs in it.

**Question 1:**

There is a template project called **Player - template** on your common drive in the following folder:

**Programming\Labsheets\labsheet 12**

You should take a ***copy*** of this template project to your drive. You should use and modify this project in order to do this question.

This project contains the following Player class:

class Player

{

// the data members are private by default

sf::RectangleShape body; // the body of the player

int speed; // the speed that the player moves

int lives; // the number of lives

int direction; // is the player moving east or west

};

The player has a rectangle body. You will need to add ***new functions*** to this class. For example you will need to add public data functions to access the private data members where required in the Player class.

There is also a **Game** class which contains a **player object** called **PlayerOne** which isa **data member** as follows:

class Game

{

// the data members are private by default

Player playerOne; // create an object of type player

public: // declaration of member functions

void loadContent();

void run();

};

You **DO NOT** need to make any changes to the Game header file.

The main( ) function within the Game.cpp file looks as follows:

int main()

{

**Game myGame;**

myGame.loadContent(); // load the font file

**myGame.run();**

system("Pause");

return 0;

}

An instance (object) of the Game class called ***myGame*** is created in the main( ) function. The Game object calls the run( ) function which contains the ***main game loop***. The player object can be initialized and used as required for the game within the run( ) function. For information on a game loop within a run() function see page 31 of the Chapter 8 notes.

There is also a **Globals.h** file. You **do not** need to make any changes to this file. All the global variables for the project are contained within this file.

**You should add the following functionality to this program:**

1. The player should have the following movement: It should move from left to right across the screen. When it reaches the boundary of the screen on the right hand side it should move from right to left. When it reaches the boundary on the left hand side it should move from left to right and so on.
2. When the user presses the ‘i’ key on the keyboard, the player’s size should increase.
3. When the user presses the ‘c’ key on the keyboard, the player’s colour should change.
4. When the user presses the ‘d’ key on the keyboard, the player should move in the opposite direction.

**Important:**

There is also an **executable** of what the program should do on your common drive within the same folder. You should run this executable. It is called **PlayerProject.exe**

**Question 2:**

You are given a **Dice** class that represents a 6 sided dice in a game. The Dice class should implement the following functions:

Dice(); // default constructor

void rollDice(); // generates a new number randomly for the dice

int getDiceNum(); // returns the dice number only

You have been given a **template** project called “Dice Game - template”. You should take a copy of this project folder onto your M drive. Modify the Dice and Game code files in this project. You **DO NOT** need to make any changes to the header files.

An object of type Dice is a data member of the Game class.

Dice gameDice; // variable to store the new Dice object

Modify the project to ***randomly*** generate another dice number when the left key is pressed. The dice number should be displayed on the screen.

The template project is on your common drive in the following folder:

**Programming\Labsheets\labsheet 12**

There is also an **executable** of what the program should do on your common drive within the same folder. You should run this executable. It is called **Dice Game.**

**Question 3:**

You have been given a **CardDeck** class that represents a deck of cards in a game. The CardDeck class should implement the following functions:

CardDeck(); // default constructor

void dealCard(); // Randomly generate a new card

// Get functions

std::string getCardSuit();//this function should only return the cardSuit

std::string getCardFace();//this function should only return the cardFace

You have been given a template project called “CardDeck - template”. You should take a copy of this project folder onto your M drive. Modify the CardDeck and Game code files in this project. You **DO NOT** need to make any changes to the header files.

An object of type CardDeck is a data member of the Game class.

CardDeck aCardDeck; // variable to store the new card desk object

Modify the project to ***randomly*** generate a new card when the left key is pressed. The number (face) of the card and suit of card should be displayed on the screen.

The template project is on your common drive in the following folder:

**Programming\Labsheets\labsheet 12**

There is also an **executable** of what the program should do on your common drive within the same folder. You should run this executable. It is called **CardDeck.**

**Question 4:**

Design a **Player SpaceShip** class suitable for a SpaceShip game in C++. The player is a type of SpaceShip and stores the game score. It provides two functions to increase and decrease the game score. The player class is represented as a rectangle that moves around the screen and is controlled via the keyboard. The player has a direction and moves in that direction within the boundary of the screen until the keyboard is used to change its direction and move in a different direction.

The player spaceship has other attributes like fuel, shield and power.

The players score is increased when it collides with the 2 enemy spaceship rectangles in the game. Design another class to represent the **Enemy SpaceShip**. The Enemy SpaceShip objects move in a direction which is randomly chosen when the object is created. They move in that direction during the game within the boundary of the screen. When they reach the side of the screen they should change to move in the opposite direction.

The Player SpaceShip objects score is displayed on the screen during the game.