Object Oriented Paradigm
Lab 14

Topic: Composition

IMPORTANT INSTRUCTIONS:

Please keep in mind the following points while coding. Violating any of these will result in credit deduction.

- There should be no memory leakage in your class. There should be no dangling pointers.
- Make functions, objects, variables as constant wherever possible.
- Create Default, Parameterized and Copy Constructor whether mentioned or not.
- Create Setters and Getters for all attributes.
- Follow the appropriate naming conventions as explained in class.

Task 1:

Many games and simulations have creatures that move around a board, map, or screen. One thing that all of these creatures have in common is that they all have a location. Task is to create a creature class that uses a point class to hold the creature's location.

First, design the point class because our creature is going to live in a 2d world, so our point class will have 2 dimensions, X and Y. We will assume the world is made up of discrete squares, so these dimensions will always be integers.

The Point class has its parts: location values x and y are part-of Point, and their lifespan is tied to that of a given Point instance. Implement the Creature class in a way that it has few properties: a name, which will be a string, and a location, which will be our Point2D class. Also add a function to move creature toward one step up, down, left or right based upon user`s input from main function.

Task 2:

Create a class Chair to store its type (char*). A room has an area and a number of Chair(s). The member functions of the Room class should have member functions addChair() and getChair(). Identify the relationship between these two classes. Create a display function in both classes. Execute the following main () function.

```
int main ()
{

        Chair c1("Plastic Chair");
        Room r1;
        r1.addChair(&c1, 1);
        c1.display();
}
```