**Lab 4 – Object Passing**

Note:

Add copy constructor and equals method in all questions.

Use this operator in copy constructor.

1. Create an encapsulated class Rectangle with **length** and **width** as data members. Create
   1. Default constructor
   2. One- argument constructor
   3. two- argument constructor
   4. display ()
   5. calculateArea ()
   6. checkSquare()
   7. CompareArea(?) //compares two objects and returns the object with larger area.
2. Create an encapsulated class Account with **balance**, **yearofOpening** and **CNIC** as data member. Create
   1. Default constructor
   2. One- argument constructor (for balance)
   3. three- argument constructor
   4. display()
   5. withdraw ()
   6. deposit()
   7. checkvalidCNIC()
3. Create an encapsulated class Point class with **x** and **y** as data members. Create
   1. Default constructor
   2. One- argument constructor (for x)
   3. two- argument constructor
   4. display()
   5. move()
   6. checkOrigin()
   7. AddTwoPoints(Point pa)// Creates and returns a new point from two other points
   8. AddThreePoints( ???)
4. Create an Encapsulated class Student with following:

Data Members:

* 1. String Name
  2. Int [] Result\_array[5] // Result array contains the marks for 5 subjects

Methods:

1. Default constructor
2. One- argument constructor (for Name)
3. two- argument constructor
4. Average ( ??? ) // it returns the average based on the marks in the array.
5. CompareAverage(?) //compares Average of two students

5. Create an Encapsulated class Book.

* 1. Its data members are
     1. author
     2. chapterNames[5]

Methods:

1. Default constructor
   1. two- argument constructor
   2. Create a method compareAuthors that compares the author of two Books and returns true if both books have same author and false otherwise. (This method Must manipulate two Book objects)
   3. Create a method compareChapters that compares the chapters of two books and returns true if both books have same chapters and false otherwise. (This method Must manipulate two Book objects)

6. Create a class “University” having following characteristics:

**Data Members:**

* String uniName;
* String rectorName,
* String location;
* String departments[20]; // it’s a string array

**Constructors:**

* No argument. {Initialize department array with five values}
* A Constructor setting values of all parameters.

**Methods:**

1. Set methods for all Data Members
2. Get methods for all data members
3. Display
   * This methods displays all the data members of the class
4. AddADepartment
   * This method should ask the user for a new department name and add it in the departments [] array.
5. CheckLocation //Boolean
   * This method should check if a university is located in a particular city or not.
6. CompareDepartment(University u )// returns the university with larger number of departments.