Data Structures - 2077
Lab Plan - 1

Exercise-1:

Write a program which calculates the area of a shape. The program should consist of two functions:

a) The first function names *GetChoice* should take the choice of input. Ask the user to input the shape for which area has to be calculated. The shapes are Square, Rectangle and Circle.
b) The second function named *CalculateArea* calculates the area according to the user choice.

Where:

*Sqaure = Radius * Radius;*

Circle = ∏* *Radius** *Radius;*

Rectangle = Length * Width;

Exercise-2:

A List stores details of 25 students (rollno, name, and marks in three subjects). Write a program to create such a list and print out a list of students who have failed in more than one subject.
[You can use structure or class concept]

Exercise-3:

Write a program which takes 10 integers from the user and sort them in ascending order. Sort the array elements using a function *AscendingSort* which sort the elements using Bubble Sort. The function takes the array as argument sorts them.

Exercise-4:

Write a class Person which has name, age and CNIC number data member. Provide a parameterized constructor of the class and a member function *Print*() which will display the data members of the class.

Exercise-5:

Edit the previously built class Person and write getter and setter functions despite of parameterized constructor of the class. The class should not be allow to set age less than 0 years and more than more than 130 years. Similarly the class does not allow to set CNIC number whose length more than or less than 13(e.g. 3312059087658) digits.

Exercise-6:

Write the definition for a class called **Distance** that has data member feet as integer and inches as float. The class has the following member functions:
void set(int, float) to give value to object
void disp() to display distance in feet and inches
Distance add(Distance) to sum two distances & return distance
1. Write the definitions for each of the above member functions.
2. Write main function to create three Distance objects. Set the value in two objects and call add() to calculate sum and assign it in third object. Display all distances.

Exercise-7:

Write a function countEven(int*, int) which receives an integer array and its size, and returns the number of even numbers in the array.

Exercise-8:

Write a function revString(char*) which reverses the parameter. The function returns nothing.

```
int main()
{
  char s[10] = "somestring";
  revString(s);  // call the function
  return 0;
}

void revtString(char* ptr)
{
  // WRITE YOUR CODE HERE
}
```

Exercise-9:

Write a program that uses pointers to swap the values of two integers entered by the user.

Exercise-10:

Create a function that takes two integers as parameters and returns their sum using pointers.

Exercise-11:

Implement a program that uses an array of pointers to store and print the names of three students entered by the user

Exercise-12:

Design a function that takes a pointer to a constant integer as a parameter and prints the value it points to

Exercise-13

Implement a set of overloaded functions that calculate the area of a square, rectangle, and circle using pointers.

Exercise-14:

Develop a program that uses pointer arithmetic to find the length of a character array entered by the user.

Exercise-15:

Write a function that accepts an array of integers and its size as parameters. The function should find and print the minimum value using pointers.