Lab	Type	Practical
Unit : I – C# Fundamentals		
1		**Variables, Data Types, Operators**
		1. Write a program to print your name, address, contact number & city.
	A	2. Write a program to get two numbers from user and print those two
	A	numbers.
		3. Write program to prompt a user to input his/her name and country name
	A	and then output will be shown as given: Hello <yourname> from country <countryname>.
		4. Write a program to calculate the size of the area in square-feet based on
	A	Specified length and width.
		5. Write a program to calculate area of Square, Rectangle and Circle.
	A	6. Write a program to calculate Celsius to Fahrenheit and vice-versa using
	B	function.
		7. Write a program to find out Simple Interest using function. (I = PRN/100)
	B	8. Write a program to create a Simple Calculator for two numbers (Addition,
	B	Multiplication, Subtraction, Division) [Also using if…else & Switch Case]
		9. Write a program to find maximum numbers from given 3 numbers.
	B	10. Write a program to Swapping without using third variable.
2		**Conditions and Looping**
	A	1. Body Mass Index (BMI) is a measure of health on weight. It can be
	A	calculated by taking your weight in kilograms and dividing by the square of your height in meters. Write a program that prompts the user to enter a weight in pounds and height in inches and displays the BMI. (Note: - 1 pound = 0.45359237 Kg and 1 inch = 0.0254 meters)
	A	2. The marks obtained by a student in 5 different subjects are input through the keyboard. The student gets a grade as per the following rules: a. Percentage above or equals to 60-first grade b. Percentage between 50 to 59-second grade c. Percentage between 40 and 49-Third grade d. Percentage less than 40-poor Grade Write a program to assign the grade obtained by the student.
	A	3. Write a program to check prime number.
	B	4. Write a program to find out whether a given year is a leap year or not.
	B	5. Write a program to print Fibonacci series.
	B	6. Write a program that takes a number as input and displays its equivalent binary form.
	B	7. Write a program to calculate the nPr. (nPr = n! / (n - r)!)

3 — **Class and Object, Constructors, Inheritance**

A	1. Write a program to create a class named **Candidate** with ID, Name, Age, Weight and Height as data members & also create a member functions like GetCandidateDetails() and DisplayCandidateDetails().
A	2. Write a program to create a class **Staff** having data members as Name, Department, Designation, Experience & Salary. Accept this data for 5 different staffs and display only names & salary of those staff who are HOD.
A	3. Write a pogram to Create a class **Bank_Account** with Account_No, Email, User_Name, Account_Type and Account_Balance as data members. Also create a Member function GetAccountDetails() & DisplayAccountDetails().
A	4. Write a program with following specifications: Class Name: **Student** Data Members: Enrollment_No, Student_Name, Semester, CPI and SPI Get Students Details using constructor and DisplayStudentDetails() using member function.
B	5. Write a program to Define a class **Salary** which will contain member variable Basic, TA, DA, HRA. Write a program using Constructor with default values for DA and HRA and calculate the salary of employee.
B	6. Write a program to Define a class **Distance** have data members dist1, dist2, dist3. Initialize the two data members using constructor and store their addition in third data member using function and display addition.
A	7. Write a program to calculate area of a Rectangle using constructor.
B	8. Create a class **Furniture** with material ,price as data members. Create another class **Table** with Height , surface_area as data members. Write a program to implement single inheritance.
A	9. Write a program for implementing single inheritance which creates one class **Account_Details** for getting account information and another class **Interest** for calculating and displaying total interest from the data inserted from account details.
B	10. Program to implement the following multiple inheritance using interface

Interface: Gross Method- Gross_sal()	Class : Salary Data Members - HRA, TA,DA Methods - Disp_sal()	Class : Employee Data Members - Name Methods - basic_sal()

4		**Exception Handling, Interface, Abstraction, String Functions**
	A	1. Write a program to Create a divide by zero exception and handle it.
	A	2. Write a program that reads 5 numbers from user. Demonstrate concept of IndexOutOfRange Exception.
	B	3. Write a program to accept a number from the user and throw an exception if the number is not an even number.
	A	4. Write a program to create an **abstract class Sum** having abstract methods SumOfTwo(int a, int b) and SumOfThree(int a, int b,int c). Create another class **Calculate** which extends the abstract class and implements the abstract methods.
	A	5. Write a program to create **interface Calculate**. In this interface we have two member functions Addition() and Subtraction(). Implements this interface in another class named Result.
	B	6. Write a program to create interface named **Shape**. In this interface, we have three methods Circle(), Triangle() and Square() which calculates area of Circle, Triangle and Square respectively. Implement Shape interface.
	A	7. Write program showing use of common methods of String class.
	A	8. Write a program to Replace lower case characters to upper case and Vice-versa.
	B	9. Write a program to find the longest word in a string.
	B	10. Write a program to change the case of entered character.

5		**Method Overloading, Method Overriding, Delegates**
	A	1. Write a program using method overloading by changing datatype of arguments to perform addition of two integer numbers and two float numbers.
	A	2. Write a program using method overloading by changing number of arguments to calculate area of square and rectangle.
	B	3. Write a programs to Find Area of Square, Rectangle and Circle using Method Overloading.
	A	4. Create a class named RBI with calculateInterest() method. Create another classes HDFC, SBI, ICICI which overrides calculateInterest() method.
	B	5. Create a class Hospital with HospitalDetails() method. Create another classes Apollo, Wockhardt, Gokul_Superspeciality which overrides HospitalDetails() method.
	A **B**	6. Write a program to return the factorial from the method using delegate.

		7. Write a program to create a delegate called TrafficDel and a class called TrafficSignal with the following delegate methods.
		Public static void Yellow()
		{
		Console.WriteLine(“Yellow Light Signal To Get Ready”);
		}
		Public static void Green()
		{
		Console.WriteLine(“Green Light Signal To Go”);
		}
		Public static void Red()
		{
		Console.WriteLine(“Red Light Signal To Stop”);
		}

	B	8. Write a program to create a delegate calculator to demonstrate the example of generic delegate.

6		**<u>Collection Classes</u>**
	A	1. Create an ArrayList for StudentName and perform following operations: a. Add() - To Add new student in list b. Remove() - To Remove Student with specified index c. RemoveRange() - To Remove student with specified range. d. Clear() - To clear all the student from the list
	A	2. Create a List for StudentName and perform following operations: a. Add() - To Add new student in list b. Remove() - To Remove Student with specified index c. RemoveRange() - To Remove student with specified range. d. Clear() - To clear all the student from the list
	B	3. Create a Stack which takes integer values and perform following operations: a. Push() - To Add new item in stack b. Pop() - To Remove item from the stack c. Peek() – To Return the top item from the stack. d. Contains() - To Checks whether an item exists in the stack or not. e. Clear() - To clear items from stack
	B	4. Create a Queue which takes integer values and perform following operations: a. Enqueue() - Adds an item into the queue. b. Dequeuer() - Returns an item from the beginning of the queue and removes it from the queue. c. Peek() - Returns an first item from the queue without removing it. d. Contains() - Checks whether an item is in the queue or not e. Clear() - Removes all the items from the queue