Course Title: **C# and Dot Net Framework**	Course code: **21BCA3C8L**
Total Contact Hours: 42	Course Credits: 03
Formative Assessment or IA Marks: 40	Duration of SEE/Exam: 02 Hours
Summative Assessment Marks: 60	

Course Outcomes (COs):

At the end of the course, students will be able to:

- Describe Object Oriented Programming concepts like Inheritance and Polymorphism in C# programming language.
- Interpret and Develop Interfaces for real-time applications.
- Build custom collections and generics in C#.

DSC8: C# and Dot Net Framework

Unit	Description	Hours
1	Introduction The C# language, the .Net Architecture and .Net Framework, The Common Language Runtime (CLR), Microsoft Intermediate Language (MSIL) Code, Just In Time Compilers (JITers),The Framework Class Library (FCL), The Common Languages Specification (CLS), The Common Type System (CTS), The .Net Framework, Working with Visual Studio.Net, Similarities and Differences between C# and C++, Java, and Visual Basic, Understanding the HELLO WORLD Application Code, The System. Environment Class, The System. Console Class, Namespaces in C#, The using Keyword, The class Keyword, The Main() Method, Printing on the Console, Comments.	10
2	C# Basics: Data Types, Variables & Constants, Operators in C#, Arithmetic Operators, Prefix and Postfix notation, Assignment Operators, Relational Operators, Other Operators, Operators precedence, Flow Control and Conditional Statements. Object and Classes: Concept of a class, Objects, Fields, Methods, Access modifiers, Properties, Static members of the class, Constructors, Destructors, Method overloading.	08
3	Pillars of OOP, Encapsulation support, Class properties, C#"s Inheritance Support, C #'s Polymorphic Support, Interface: Deriving classes, calling base class constructor, Overriding Methods, Non-Inheritable Classes, Abstract Class, Interface Inheritance, Namespace and Access Modifiers, Boxing and Un-boxing. .NET Delegate type, defining a Delegate in C#, System. Delegate Base Classes, Delegate examples, C# Events, operator overloading.	08
4	Exception Handling: Handling Exceptions using try and catch, Raising Exceptions using throw, Pre- defined Exception classes, Custom Exception classes, Understanding Object Lifetime classes, Objects and References, the basics of Object Lifetime, System. GC type. Assemblies-The Role of .NET Assemblies, Understanding the format of .NET Assemblies, single file assembly, multifile assembly, Private and Shared Assemblies.	08

5	Working with Collections: List and Dictionary, Array List and Hash Table, Generic Classes, Comparable and Sorting, WinForms: Introduction, Controls, Menus and Context Menus, Menu Strip, Toolbar Strip, Graphics and GDI, SDI and MDI Applications, Dialog box (Modal and Modeless), Form Inheritance, Developing Custom, Composite and Extended Controls.	08

References:

1. E. Balagurusamy, Programming in C#, Tata McGraw Hill
2. Stephen Walthert, ASP.NET 3.5 unleased, SAMS
3. ShibiPanikkar and Kumar Sanjeev, C# with .NET Frame Work, Firewall Media
4. Jeffrey Richter, Applied Microsoft .Net Framework Programming, (Microsoft)

Additional Reading:

5. http://www.bestdotnettraining.com
6. http://www.bestsharepointtraining.com
7. https://stackoverflow.com/documentation
8. Troelsen, Andrew, Pro C# 5.0 and the .NET 4.5 Framework, 6th Edition, APress, India

Year	II	**Course Code: 21BCA3C8P**	**Credits**	02
Sem.	III	**Course Title: C# and Dot Net Framework LAB**	**Hours**	40
Course Pre-requisites, ifany:		Knowledge of Programming		
Formative Assessment Marks: 25		Summative Assessment Marks: 25	Duration of ESA: 03 hrs.	

Practicals:

1. Write a C# program to show the machine details like machine name, Operating System, Version, Physical Memory and calculate the time since the Last Boot Up. (Hint: Use System. Environment Class)
2. Write a program in C# Sharp to count a total number of alphabets, digits and special characters in a string
3. Write a program in C# Sharp to create a function to calculate the sum of the individual digits of a given number.
4. Draw a square with sides 100 pixels in length. Then inscribe a circle of radius 50 inside the square. Position the square and the inscribed circle in the middle of the screen.
5. Write a program to implement multilevel inheritance.
6. Write a program to demonstrate System exception.
7. Write an object oriented program to demonstrate bank transaction. Include methods for amount deposit, amount withdrawal and display.
8. Demonstrate operator overloading two complex numbers.
9. Demonstrate Dialog box (Modal and Modeless).
10. Write a program in C# Sharp to create Menu and menu items.

Evaluation Scheme for Lab Examination:

Assessment Criteria		**Marks**
Program – 1 from Part A	Writing the Program	03
	Execution and Formatting	07
Program -2 from Part B	Writing the Program	03
	Execution and Formatting	07
Viva Voice based on **C# and Dot Net Framework**		05
Total		25