	Samuel Control	SKIT	Teaching Process	Rev No.: 1.0
		Doc Code:	SKIT.Ph5b1.F02	Date: 27-02-2019
160		Title:	Course Plan	Page: 1 / 27

\_

# Table of Contents

15CS564: DOT NET FRAMEWORK	2
A. COURSE INFORMATION	2
1. Course Overview	2
2. Course Content	2
3. Course Material	2
4. Course Prerequisites	3
B. OBE PARAMETERS	3
1. Course Outcomes	
2. Course Applications	4
3. Articulation Matrix	4
4. Mapping Justification	
5. Curricular Gap and Content	88
6. Content Beyond Syllabus	
C. COURSE ASSESSMENT	88
1. Course Coverage	
2. Continuous Internal Assessment (CIA)	
D1. TEACHING PLAN - 1	
Module - 1	
Module – 2	
E1. CIA EXAM – 1	
a. Model Question Paper - 1	
b. Assignment -1	
D2. TEACHING PLAN - 2	
Module – 3	
Module – 4	
E2. CIA EXAM – 2	
a. Model Question Paper - 2	
b. Assignment – 2	
D3. TEACHING PLAN - 3	
Module – 5	
E3. CIA EXAM – 3	
a. Model Question Paper - 3	
b. Assignment – 3	
F. EXAM PREPARATION	
1. University Model Question Paper	17
2 SEE Important Questions	19

Note: Remove "Table of Content" before including in CP Book

Souther Control	SKIT	Teaching Process	Rev No.: 1.0
	Doc Code:	SKIT.Ph5b1.F02	Date: 27-02-2019
	Title:	Course Plan	Page: 2 / 27

Each Course Plan shall be printed and made into a book with cover page Blooms Level in all sections match with A.2, only if you plan to teach / learn at higher levels

## 15CS564: DOT NET FRAMEWORK

#### A. COURSE INFORMATION

#### 1. Course Overview

Degree:	BE	Program:	CS
Semester :	V	Academic Year:	2018-19
Course Title:	DOT NET FRAMEWORK	Course Code:	15CS564
Credit / L-T-P:	3-0-0	SEE Duration:	180 Minutes
Total Contact Hours:	40	SEE Marks:	80 Marks
CIA Marks:	20	Assignment	3
Course Plan Author:	Vamsi Krishna Y	Sign	Dt: 15-09-2018
Checked By:		Sign	Dt:

#### 2. Course Content

Mod	Module Content	Teaching	Module	Blooms
ule		Hours	Concepts	Level
1	Introducing Microsoft Visual C# and Microsoft Visual Studio 2015: Welcome to C#, Working with variables, operators and expressions, Writing methods and applying scope, Using decision statements, Using compound assignment and iteration statements, Managing errors and exceptions		syntax and semantics	L2
2	Understanding the C# object model: Creating and Managing classes and objects, Understanding values and references, Creating value types with enumerations and structures, Using arrays		Object Oriented concepts	L3
3	<b>Understanding parameter arrays:</b> Working with inheritance, Creating interfaces and defining abstract classes, Using garbage collection and resource management		Event Handling, Custom Interface	L6
4	<b>Defining Extensible Types with C#:</b> ,Implementing properties to access fields, Using indexers, Introducing generics, Using collections		properties, index's, generics and collections	L3
5	<b>Enumerating Collections:</b> Decoupling application logic and handling events, Querying in-memory data by using query expressions, Operator overloading		operator with behavior, queries to query in- memory	L6

Samuel Control	SKIT	Teaching Process	Rev No.: 1.0
	Doc Code:	SKIT.Ph5b1.F02	Date: 27-02-2019
	Title:	Course Plan	Page: 3 / 27

### 3. Course Material

Mod	Details	Available
ule		
1	Text books	
	John Sharp, Microsoft Visual C# Step by Step, 8th Edition, PHI Learning Pvt. Ltd. 2016	In Lib
		In Lib
2	Reference books	
	1.Christian Nagel, "C# 6 and .NET Core 1.0", 1st Edition, Wiley India Pvt Ltd, 2016.	In Lib
	2Andrew Stellman and Jennifer Greene, "Head First C#", 3rd Edition, O'Reilly Publications, 2013.	In Lib
	3Andrew Troelsen, "Prof C# 5.0 and the .NET 4.5 Framework", 6th Edition, Apress and Dreamtech Press, 2012.	In Lib
3	Others (Web, Video, Simulation, Notes etc.)	
		Not Available
	https://www.youtube.com/watch?v=akEr8cUAd5g	

# 4. Course Prerequisites

SNo	Course Code	Course Name	Module / Topic / Description	Sem	Remarks	Blooms Level
1		Concepts	A Review of structures, Procedure—Oriented Programming system, Object Oriented Programming System, Console I/O, variables and reference variables, Function Prototyping, Function Overloading. Class and Objects Introduction, member functions and data, objects and functions, objects and arrays,			L3

Souther Co.	SKIT	Teaching Process	Rev No.: 1.0
	Doc Code:	SKIT.Ph5b1.F02	Date: 27-02-2019
	Title:	Course Plan	Page: 4 / 27

Note: If prerequisites are not taught earlier, GAP in curriculum needs to be addressed. Include in Remarks and implement in B.5.

#### **B. OBE PARAMETERS**

#### 1. Course Outcomes

- N.I I. I	Total	40	-	-	-	-
15CS564.8	Design queries to query in- memory data	4	Queries of Data		Seminar and assignment	L6
15CS564.7	Develop own operator with behavior	4	Operator	discussi on	9	L3
6	Illustrate generics and collections	4	generics and collections	discussio n	Assignment, viva	L3
	Discuss the concepts of properties, index's	4	Properties and index's	Discussio n, lecture, ppt	Presentation , assignment	L2
15CS564.4	Design custom interfaces	4	Interface (built in ,Custom )	n	Question and answer, test	L6
	Apply Event Handling mechanism in the application	4	Event Handling and Garbage Collection	Lecture / PPT, problem solving	Assignment, seminar	L3
	Demonstrate Object Oriented concepts in C#	8	Basics of OOC	Lecture / PPT,	Assignment, seminar	L3
	understanding the syntax and semantics of C# Application	8	Basic Concepts	Lecture, discussio n	Assignment	L2
#	COs	Teach. Hours	Concept	Instr Method	Assessment Method	Blooms' Level

Note: Identify a max of 2 Concepts per Module. Write 1 CO per concept.

#### 2. Course Applications

SNo	Application Area	CO	Level		
1	Stand alone C# applications	CO1	L2		
2					
3	Menu based banking application	CO3	L3		
4	Scientific application with custom interface real time	CO4	L6		
5	File structure applications	CO5	L2		
6	Standalone Application with garbage collections	CO6	L3		
7	Reusable application	CO7	L3		
8	Database application with real data	CO8	L6		

Commercial	SKIT	Teaching Process	Rev No.: 1.0
	Doc Code:	SKIT.Ph5b1.F02	Date: 27-02-2019
	Title:	Course Plan	Page: 5 / 27

Copyright ©2017. cAAS. All rights reserved.

Note: Write 1 or 2 applications per CO.

### 3. Articulation Matrix

_	Course Outcomes	Program Outcomes												
#	COs	PO <sub>1</sub>	PO2	PO3	PO4	PO5	P06	PO7	P08	РО	РО	PO <sub>1</sub>	PO <sub>1</sub>	Level
										9	10	1	2	
15CS564.1	understanding the syntax and semantics of C# Application	√				V								L2,L3
15CS564.2	Demonstrate Object Oriented concepts in C#	√	√			V								L3
15CS564.3	Apply Event Handling mechanism in the application	1	<b>√</b>			√								L3,L4
15CS564.4	Design custom interfaces	√	√	√	√	√			√	√	√			L6
	Discuss the concepts of properties, index's	√	√			√								L2,L3
15CS564.6	Illustrate generics and collections	1	1	1		1								L3
15CS564.7	Develop own operator with behavior	1	√	1		1								L3,L4
15CS564.8	Design queries to query in- memory data	1	1	1	1	1				1				L6
Note: Menti	on the mapping strength as 1, 2,	or 3												

Commerce of the Commerce of th	SKIT	Teaching Process	Rev No.: 1.0
	Doc Code:	SKIT.Ph5b1.F02	Date: 27-02-2019
	Title:	Course Plan	Page: 6 / 27

## 4. Mapping Justification

Sommer State of the State of th	SKIT	Teaching Process	Rev No.: 1.0
	Doc Code:	SKIT.Ph5b1.F02	Date: 27-02-2019
	Title:	Course Plan	Page: 7 / 27

Copyright ©2017. cAAS. All rights reserved Justification Mapping Mapping Level PO CO PO1 CO<sub>1</sub> Yes Mapping ( Requires the basic Engineering core knowledge but at the remembrance level ) PO2 Not Required (it is restricted to remembrance not at the level of analysis) PO<sub>3</sub> No Mapping (Learning at basic level no design or development of solutions) PO<sub>4</sub> No Mapping (No investigation or interpretation of content) PO5 Yes Mapping (modern tool Is used ,visual studio) P06 No Mapping ( No social cultural issues with syntax and semantics) PO7 No Mapping ( No impact on Environmental and sustainability issues with respect to the syntax and semantics ) PO8 No Mapping (since there is no team work or lead, for the ethical work) PO9 No Mapping (since there is no team work or lead) PO10 No Mapping ( as there is no communication , the level expected Is remembering) PO11 No Mapping (as there Is no communication for project and finance, the level expected is remembering) PO12 No Mapping (as there Is only understanding and remembering ,there Is no change in syntax ) CO<sub>2</sub> PO<sub>1</sub> Yes Mapping (Requires the basic Engineering core knowledge of OOC but at the remembrance level) PO<sub>2</sub> Yes Mapping (Requires the analysis of OOC in reusable applications at understanding level ) PO<sub>3</sub> No Mapping (Learning at basic level no design or development of solutions) PO<sub>4</sub> No Mapping ( No investigation or interpretation of content ) PO5 Yes Mapping (modern tool Is used ,visual studio) P06 No Mapping ( No social cultural issues with under of the syntaz and semantics) PO7 No Mapping ( No impact on Environmental and sutainability issues with under of the syntaz and semantics ) PO8 No Mapping (since there is no team work or lead for the ethical work) PO9 No Mapping ( since there is no team work or lead ) PO10 No Mapping ( as there Is no communication , the level expected Is remembering) PO11 No Mapping (as there Is no project development involved with respect to **ooc** at the remembrance level, the level expected Is remembering) PO12 No Mapping (as there Is only understanding and remembering of ooc, there is no life long learning) CO<sub>3</sub> PO<sub>1</sub> Yes Mapping (Reguires the basic Engineering core knowledge of OOC and C#, but at the remembrance level) PO<sub>2</sub> Yes Mapping (Requires the analysis of Error handling mechanic in different applications) PO3 No Mapping (Learning at basic level no design or development of solutions)

	SKIT	Teaching Process	Rev No.: 1.0
	Doc Code:	SKIT.Ph5b1.F02	Date: 27-02-2019
Copyright ©2017. cA	Title:	Course Plan	Page: 8 / 27
,·-g	PO4	No Mapping ( No investigation or interpretation of content )	
	PO5	Yes Mapping (modern tool Is used ,visual studio)	
	PO6	No Mapping ( No social cultural issues with under of the and semantics )	syntaz
	P07	No Mapping ( No impact on Environmental and sutainabilitiesues with under of the syntaz and semantics )	<u>У</u>
	PO8	No Mapping ( since there is no team work or lead for the work )	ethical
	PO9	No Mapping ( since there is no team work or lead )	
	PO10	No Mapping ( as there Is no communication , the level expected Is remembering )	
	PO11	No Mapping ( as there Is no communication , the level expected Is remembering )	
	PO12	No Mapping ( as there Is only understanding and remembering )	
CO4	PO1	Yes Mapping ( Requires the basic Engineering core known of OOC and C#, but at the remembrance level )	wledge
	PO2	Yes Mapping ( Requires the analysis of Error handling mecthehanic in different applications)	
	PO3	Yes Mapping (design of GUI requires the customized application)	
	PO4	Yes Mapping (Deep investigation and interpretation of correquired in different application development for Interface	
	PO5	Yes Mapping ( modern tool Is used ,visual studio)	
	PO6	No Mapping ( No social cultural issues with under of the and semantics )	syntaz
	PO7	No Mapping ( No impact on Environmental and sutainabilities uses with under of the syntaz and semantics )	у
	PO8	Yes Mapping ( since there is a team work required for requirement gathering in designing the interface, the principals are important)	
	PO9	Yes Mapping ( since there is a team work required for requirement gathering in designing the interface )	
	PO10	Yes Mapping ( communication is required as there is a twork involved )	:eam
	PO11	No Mapping ( as there Is no communication , the level expected Is remembering )	
	PO12	No Mapping ( as there Is only understanding and remembering )	
CO5	PO1	Yes Mapping (Requires the basic Engineering core known of OOC and C#, but at the remembrance level)	wledge
	PO2	Yes Mapping (Requires the analysis of properties and different web applications)	index in
	PO <sub>3</sub>	No Mapping (Learning at basic level no design or develo of solutions)	pment
	PO4	No Mapping ( No investigation or interpretation of content )	
	PO5	Yes Mapping ( modern tool Is used ,visual studio)	
	P06	No Mapping ( No social cultural issues with concepts of	:
	D.0	properties and index's)	
	PO7	No Mapping ( No impact on Environmental and sutainabilitiessues with concepts of properties and index's )	
	PO8	No Mapping ( since there is no team work or lead for the	ethical

	SKIT	Teaching Process	Rev No.: 1	.О			
	Doc Code:	SKIT.Ph5b1.F02	Date: 27-0	)2-2019			
	Title:	Course Plan	Page: 9 /	27			
Copyright ©2017. cAAS. All rights reserved.							
		work)					

No.	SKIT	Teaching Process	Rev No.: 1.0
	Doc Code:	SKIT.Ph5b1.F02	Date: 27-02-2019
Copyright ©2017 cA	Title: .AS. All rights reserve	Course Plan	Page: 10 / 27
oopyngni @2017. G/	PO9	No Mapping ( since there is no team work or lead )	
	PO10	No Mapping ( as there Is no communication , the level	
	DO	expected Is remembering )	
	PO11	No Mapping ( as there Is no communication , the level expected Is remembering )	
	PO12	No Mapping ( as there Is only understanding and remembering )	
CO6	PO1	Yes Mapping (Requires the basic Engineering core known of OOC and C#, but at the remembrance level)	wledge
	PO2	Yes Mapping (Requires the analysis of generics and collection in File Structure applications)	
	PO3	yes Mapping (Learning at basic level of design and development is required in generics and collections v simulating the collections )	while
	PO4	No Mapping ( No investigation or interpretation of content )	)
	PO5	Yes Mapping ( modern tool Is used ,visual studio)	
	P06	No Mapping ( No social cultural issues with concepts of properties and index's)	f
	PO7	No Mapping ( No impact on Environmental and sutainabilissues with concepts of properties and index's )	ty
	PO8	No Mapping ( since there is no team work or lead for the work )	e ethical
	PO9	No Mapping ( since there is no team work or lead )	
	PO10	No Mapping ( as there Is no communication , the level expected Is remembering )	
	PO11	No Mapping ( as there Is no communication , the level expected Is remembering )	
	PO12	No Mapping ( as there Is only understanding and remembering )	
CO7	PO1	Yes Mapping (Requires the basic Engineering core known of OOC and C#, but at the remembrance level)	wledge
	PO2	Yes Mapping (Requires the analysis of operators)	
	PO <sub>3</sub>	yes Mapping (Learning at basic level of design and development is required in operators and with its overloading )	
	PO4	No Mapping ( No investigation or interpretation of content )	)
	PO <sub>5</sub>	Yes Mapping (modern tool Is used ,visual studio)	
	P06	No Mapping ( No social cultural issues with concepts of properties and index's)	f
	PO7	No Mapping ( No impact on Environmental and sutainabili issues with concepts of properties and index's )	ty
	PO8	No Mapping ( since there is no team work or lead for the work )	e ethical
	PO9	No Mapping ( since there is no team work or lead )	
	PO10	No Mapping ( as there Is no communication , the level expected Is remembering )	
	PO11	No Mapping ( as there Is no communication , the level expected Is remembering )	
	PO12	No Mapping ( as there is only understanding and remembering )	
CO8	PO1	Yes Mapping (Requires the basic Engineering core known of OOC and C#, but at the remembrance level)	wledge

Town or the second	SKIT	Teaching Process R	Rev No.: 1.0
(200)	Doc Code:		ate: 27-02-2019
	Title:		age: 11 / 27
Copyright ©2017. c	AAS. All rights reserve	d.	
	PO2	Yes Mapping (Requires the analysis of memory handling different applications)	g in
	PO3	Yes Mapping (design of queries in time and memory management of customized application)	
	PO4	Yes Mapping (Deep investigation and interpretation of conterequired in memory handling with respect to the queries different application development for Interface's)	
	PO5	Yes Mapping (modern tool Is used ,visual studio)	
	P06	No Mapping ( No social cultural issues with under of the sand semantics )	syntaz
	PO7	No Mapping ( No impact on Environmental and sutainability issues with under of the syntaz and semantics )	
	PO8	No Mapping ( since there is no team work or lead for the e work )	thical
	PO9	yes Mapping (since there is a multidisciplinary settings in memory management)	
	PO10	No Mapping (communication is not required as there is r team work involved)	10
	PO11	No Mapping ( as there Is no communication , the level expected Is remembering )	
	PO12	No Mapping ( as there Is only understanding and remembering )	

remembering )

Note: Write justification for each CO-PO mapping.

### 5. Curricular Gap and Content

SNo	Gap Topic	Schedule Planned	Resources Person	PO Mapping
1	15CS45 / Object Oriented Concepts			
2				
3				
4				
5				

Note: Write Gap topics from A.4 and add others also.

## 6. Content Beyond Syllabus

SNo	Gap Topic	Actions Planned	Schedule Planned	Resources Person	PO Mapping
1					
2					
3					
4					
5					

Samuel Control	SKIT	Teaching Process	Rev No.: 1.0
	Doc Code:	SKIT.Ph5b1.F02	Date: 27-02-2019
	Title:	Course Plan	Page: 12 / 27

Note: Anything not covered above is included here.

#### C. COURSE ASSESSMENT

#### 1. Course Coverage

Mod	Title	Teaching		No. o	f quest	ion in	Exam		CO	Levels
ule		Hours	CIA-1	CIA-2	CIA-3	Asg	Extra	SEE		
#							Asg			
	Introducing Microsoft Visual C# and	8	2	_	-	1	1	2	CO1	L2
	Microsoft Visual Studio 2015									
2	Understanding the C# object	8	2	_	-	1	1	2	CO2	L3
	model									
3	Understanding parameter arrays	8	-	2	-	1	1	2	CO3,	L3,L6
									CO4	
4	Defining Extensible Types with C#	8	-	2	2	1	1	2	CO5,	L2,L3
									CO6	
5	Enumerating Collections	8	-	-	2	1	1	2	C07,	L3,L6
	_								CO8	
-	Total	40	4	4	4	5	5	10	-	-

Note: Distinct assignment for each student. 1 Assignment per chapter per student. 1 seminar per test per student.

#### 2. Continuous Internal Assessment (CIA)

Evaluation	Weightage in Marks	CO	Levels
CIA Exam – 1	15	CO1, CO2	L2,L3
CIA Exam – 2	15	CO3,CO4,CO5, CO6	L3,L6, L2, L3
CIA Exam – 3	15	CO5,CO6,CO7, CO8	L2,L3,L3 L6
Assignment - 1	05	CO1, CO2	L2,L3
Assignment - 2	05	CO3,CO4,CO5, CO6	L3,L6, L2,L3
Assignment - 3	05	CO5,CO6,CO7, CO8	L2,L3,L3,L6
Seminar - 1			
Seminar - 2			
Seminar - 3			
Other Activities – define –			
Slip test			
Final CIA Marks	20	-	-

South 1	SKIT	Teaching Process	Rev No.: 1.0
	Doc Code:	SKIT.Ph5b1.F02	Date: 27-02-2019
	Title:	Course Plan	Page: 13 / 27

Note: Blooms Level in last column shall match with A.2 above.

### D1. TEACHING PLAN - 1

#### Module - 1

Title:	Introducing Microsoft visual c#	Appr Time:	8 Hrs
a	Course Outcomes	-	Blooms
-	The student should be able to:	-	Level
1	understanding the syntax and semantics of C# Application	CO1	L2
b	Course Schedule	-	-
Class N	o Module Content Covered	СО	Level
1	Introducing Microsoft Visual C# and Microsoft Visual Studio 2015	C01	L2
2	Welcome to C#	C01	L2
3	Working with variables, operators and expressions	C01	L2
4	Writing methods and applying scope	C01	L2
5	Using decision statements	C01	L2
6	Using compound assignment	C01	L2
7	Using iteration statements	C01	L2
8	Managing errors and exceptions	C01	L2
С	Application Areas	СО	Level
1	Stand alone C# applications	CO1	L2
2	OOPs based programming	CO1	L2
d	Review Questions	-	-
1	What is Namespace? How namespace is used to solve name-clashing problem, Explain with Example.	CO1	L2
	problem, Explain with Example.	CO1	L2 L2
1 2 3	problem, Explain with Example.  What is an assembly? Explain with Examples  Explain how to create a blank universal windows platform app using visual		
2	problem, Explain with Example.  What is an assembly? Explain with Examples  Explain how to create a blank universal windows platform app using visual studio 2015 with example.	CO1	L2 L2
2	problem, Explain with Example.  What is an assembly? Explain with Examples  Explain how to create a blank universal windows platform app using visual studio 2015 with example.  Discus a C# console application  Explain how the precedence and associativity of operators determine and	CO1	L2
2 3 4	problem, Explain with Example.  What is an assembly? Explain with Examples  Explain how to create a blank universal windows platform app using visual studio 2015 with example.  Discus a C# console application  Explain how the precedence and associativity of operators determine and how expressions are evaluated in C#.  Explain how to write methods that take optional parameters and how to	CO1 CO1	L2 L2
2 3 4 5	problem, Explain with Example.  What is an assembly? Explain with Examples  Explain how to create a blank universal windows platform app using visual studio 2015 with example.  Discus a C# console application  Explain how the precedence and associativity of operators determine and how expressions are evaluated in C#.  Explain how to write methods that take optional parameters and how to call methods by using named parameters.  Explain how to use the compound assignment operators to update	CO1 CO1 CO1	L2 L2 L2
2 3 4 5	problem, Explain with Example.  What is an assembly? Explain with Examples  Explain how to create a blank universal windows platform app using visual studio 2015 with example.  Discus a C# console application  Explain how the precedence and associativity of operators determine and how expressions are evaluated in C#.  Explain how to write methods that take optional parameters and how to call methods by using named parameters.	CO1 CO1 CO1 CO1	L2 L2 L2 L2 L2
2 3 4 5 6	problem, Explain with Example.  What is an assembly? Explain with Examples  Explain how to create a blank universal windows platform app using visual studio 2015 with example.  Discus a C# console application  Explain how the precedence and associativity of operators determine and how expressions are evaluated in C#.  Explain how to write methods that take optional parameters and how to call methods by using named parameters.  Explain how to use the compound assignment operators to update numeric variables and append one string to another  Explain how to use while, for, and do statements to execute code	CO1 CO1 CO1 CO1 CO1	L2 L2 L2 L2 L2
2 3 4 5 6 7 8	problem, Explain with Example.  What is an assembly? Explain with Examples  Explain how to create a blank universal windows platform app using visual studio 2015 with example.  Discus a C# console application  Explain how the precedence and associativity of operators determine and how expressions are evaluated in C#.  Explain how to write methods that take optional parameters and how to call methods by using named parameters.  Explain how to use the compound assignment operators to update numeric variables and append one string to another  Explain how to use while, for, and do statements to execute code repeatedly while some Boolean condition is true with example  Explain the two things you need write exception-aware programs. Explain with example how to catch and handle exceptions by using the try and	CO1 CO1 CO1 CO1 CO1 CO1	L2 L2 L2 L2 L2 L2 L2
2 3 4 5 6 7 8	problem, Explain with Example.  What is an assembly? Explain with Examples  Explain how to create a blank universal windows platform app using visual studio 2015 with example.  Discus a C# console application  Explain how the precedence and associativity of operators determine and how expressions are evaluated in C#.  Explain how to write methods that take optional parameters and how to call methods by using named parameters.  Explain how to use the compound assignment operators to update numeric variables and append one string to another  Explain how to use while, for, and do statements to execute code repeatedly while some Boolean condition is true with example  Explain the two things you need write exception-aware programs. Explain with example how to catch and handle exceptions by using the try and	CO1 CO1 CO1 CO1 CO1 CO1	L2 L2 L2 L2 L2 L2 L2
2 3 4 5 6 7 8 9	problem, Explain with Example.  What is an assembly? Explain with Examples  Explain how to create a blank universal windows platform app using visual studio 2015 with example.  Discus a C# console application  Explain how the precedence and associativity of operators determine and how expressions are evaluated in C#.  Explain how to write methods that take optional parameters and how to call methods by using named parameters.  Explain how to use the compound assignment operators to update numeric variables and append one string to another  Explain how to use while, for, and do statements to execute code repeatedly while some Boolean condition is true with example  Explain the two things you need write exception-aware programs. Explain with example how to catch and handle exceptions by using the try and catch constructs	CO1 CO1 CO1 CO1 CO1 CO1	L2 L2 L2 L2 L2 L2 L2

Sommer Control	SKIT	Teaching Process	Rev No.: 1.0
	Doc Code:	SKIT.Ph5b1.F02	Date: 27-02-2019
	Title:	Course Plan	Page: 14 / 27

### Module - 2

Title:	Understanding C# object model	Appr Time:	8 Hrs
a	Course Outcomes	_	Blooms
-	The student should be able to:	-	Level
1	Demonstrate Object Oriented concepts in C#	CO <sub>2</sub>	L3
b	Course Schedule	-	-
lass N	Module Content Covered	СО	Level
1	Understanding the C# object model.	CO2	L2
2	Creating and Managing classes and objects.	CO2	L3
3	Creating and Managing classes and objects.	CO2	L3
4	Understanding values and references	CO2	L3
5	Understanding values and references	CO2	L2
6	Creating value types with enumerations and structures	CO2	L2
7	Creating value types with enumerations and structures	CO2	L2
8	Using arrays	CO2	L2
С	Application Areas	СО	Level
1	Array based application	CO2	L3
2	Stand alone algorithm application	CO2	L2
d	Review Questions	-	-
1	Explain the purpose of encapsulation with example.	CO2	L2
2	What is constructor? Explain necessity of overloading constructors with example.	CO2	L1
3	Explain how we create a with example  1) Sharing field 2) static field by using the const keyword  3)Static Class 4) Anonymous classes.	CO2	L2
4	Explain the differences between a value type and a reference type with example.	CO2	L2
5	Apply the concept of boxing and unboxing? Explain with examples.	CO2	L3
6	Explain how arguments are passed as method parameters by using the ref and out keywords	CO2	L2
7	Explain how to control the accessibility of members by using the public and private keywords with examples.	CO2	L2
8	Write and call your own constructors in C#. Explain how to create anonymous classes with examples.	CO2	L2
е	Experiences		
1			
2			

Samuel Control	SKIT	Teaching Process	Rev No.: 1.0
	Doc Code:	SKIT.Ph5b1.F02	Date: 27-02-2019
	Title:	Course Plan	Page: 15 / 27

### E1. CIA EXAM – 1

## a. Model Question Paper - 1

Crs (	ode.	15CS564	Sem:	V	Marks:	30	Time: 7	5 minute	)C	
Cour		DOT NET F		•	i lano.	50	THITIE.	Jimilate	.5	
-					each carry e	qual marl	ks.	Marks	СО	Level
1						-	riting identifiers. Li	st 9	CO1	L2
		out the diff	t the different keywords							
	b	Explain the	different pr	rimitive's o	data types wit	h size, rar	nge and example fo	or 6	CO1	L2
		each	·							
					OR					
2	а	Explain the	concept of	creating n	nethods with	example		9	CO1	L2
	b	Explain app	lying scope	for local s	cope and clas	s scope w	ith examples.	6	CO1	L2
3	а	Explain the	different ar	ithmetic c	perators with	example	S.	5	CO1	L2
	b	What are va	ariables? Lis	t out the i	rules for nami	ng variabl	les.	5	CO1	L1
	С	Explain the	optional pa	rameters	and named ar	gument w	vith examples.	5	CO1	L2
					OR					
4	а	Explain the	Boolean op	erators w	ith examples.			8	CO1	L2
	b	Explain the	different de	ecision ma	king statemer	nts.		7	CO1	L2

Commerce of the Commerce of th	SKIT	Teaching Process	Rev No.: 1.0
	Doc Code:	SKIT.Ph5b1.F02	Date: 27-02-2019
	Title:	Course Plan	Page: 16 / 27

### b. Assignment -1

Note: A distinct assignment to be assigned to each student.

	Model Assignment Questions									
Crs C	ode:	15CS564	ßem:	V	Marks:	5 / 10	Time:	90 – 120	minute	S
Cours	se:									
			to answer	2-3 assign	ments. Each as	ssignment c	arries equal m	nark.		
SNo	U	USN		Α	ssignment De	scription		Marks	СО	Level
1					? How namesp		l to solve nam	e- 8	CO1	L2
			<u> </u>		xplain with Exa	•				
2					/? Explain with			4	CO1	L2
3					te a blank unive		ws platform a <sub>l</sub>	op 6	CO1	L2
					015 with exam <sub>l</sub>	ole.				
4					e application			4	C01	L2
5			determine	and how	ecedence and expressions ar	e evaluated	l in C#.		C01	L2
6					methods that hods by using I			6	C01	L2
7					he compound ables and appe			5	C01	L2
8			Explain ho	ow to use v	while, for, and c ile some Boole	lo statemer	nts to execute	6	C01	L2
9			Explain th programs	. Explain w	gs you need wi ith example ho the try and cat	w to catch	and handle	6	C01	L2
10			Explain th	e purpose	of encapsulati	on with exa	mple.	4	CO2	L2
11			What is co		Explain neces			5	CO2	L1
12				1) Sharing 1	ate a with exam field 2) static fie )Static Class 4)	id by using		8	CO2	L2
13			Explain th type with		es between a	value type a	and a referenc	e 4	CO2	L2
14			Apply the examples		of boxing and u	nboxing? E>	kplain with	4	CO2	L3
15			Explain ho	w argume	ents are passec t keywords	l as method	parameters b	У	CO2	L2
16			Explain ho	w to conti	rol the accessik te keywords wi				CO2	L2
17			Write and	call your o	own constructo	ors in C#. Ex			CO2	L2

Sommer Con	SKIT	Teaching Process	Rev No.: 1.0
	Doc Code:	SKIT.Ph5b1.F02	Date: 27-02-2019
	Title:	Course Plan	Page: 17 / 27

## D2. TEACHING PLAN - 2

# Module - 3

Title:	Parameterized Arrrays	Appr Time:	8 Hrs
а	Course Outcomes	-	Blooms
-	The student should be able to:	-	Level
1	Apply Event Handling mechanism in the application	CO3	L3
2	Design custom interfaces	CO4	L6
b	Course Schedule		
Class No	Module Content Covered	CO	Level
1	Understanding parameter arrays	CO3	L3
2	Working with inheritance	CO3	L3
3	Working with inheritance	CO3	L3
4	Working with inheritance	CO3	L3
5	Creating interfaces and defining abstract classes	CO4	L6
6	Creating interfaces and defining abstract classes	CO4	L6
7	Using garbage collection	CO4	L3
8	Using resource management	CO <sub>4</sub>	L3
С	Application Areas	СО	Level
1	Network applications	CO3	L3
2	Windows form based applications	CO4	L6
d	Review Questions	-	-
1	Write a method that can accept any number of arguments by using the params keyword.	CO3	L1
2	Discus method overloading? Explain with example.	CO3	L2
3	Explain how to create a derived class that inherits features from a base class with example.	CO3	L2
4	Explain how to implement an interface in a structure or class with examples.	CO3	L2
5	How to limit accessibility within an inheritance hierarchy by using the protected keyword, explain with example.	CO3	L6
6	Define extension methods as an alternative mechanism to using inheritance with examples	CO4	L2
7	Define an interface specifying the signatures and return types of methods with examples	CO4	L2
8	Explain how to implement an interface in a structure or class with examples	CO4	L2
9	Explain how to manage system resource by using garbage collection with example	CO4	L2
10	Write code in c# that runs when an object is destroyed	CO4	L6
е	Experiences	_	-
1			

Commercial	SKIT	Teaching Process	Rev No.: 1.0
	Doc Code:	SKIT.Ph5b1.F02	Date: 27-02-2019
	Title:	Course Plan	Page: 18 / 27

## Module - 4

Title:	Defining Extensible Types with C#	Appr Time:	16 Hrs
a	Course Outcomes	-	Blooms
-	The student should be able to:	-	Level
1	<b>Discuss</b> the concepts of properties, index's	CO5	L2
2	Illustrate generics and collections	CO6	L3
	<u> </u>		
b	Course Schedule		
Class No	Module Content Covered	СО	Level
1	Defining Extensible Types with C#	CO5	L2
2	Implementing properties to access fields	CO5	L2
3	Implementing properties to access fields	CO5	L2
4	Using indexers	CO5	L2
5	Introducing generics	CO6	L3
6	Introducing generics	CO6	L3
7	Using collections	CO6	L3
8	Using collections	CO6	L3
	Application Areas	СО	Level
<b>C</b>	Application Areas Custom control applications	CO6	Level L2
t	Review Questions	-	
1	Discuss the properties? Explain how to create and use properties to provide controlled access to data in an object with examples.	CO <sub>5</sub>	L2
2	Explain how to control read access to properties by declaring get accessors with example	CO5	L2
3	Demonstrate interfaces in c# containing properties by using structures and classes.	CO <sub>5</sub>	L3
4	Explain how to control write access to properties by declaring set accessors with example	CO5	L2
5	Explain how to create interfaces that declare properties with example	CO5	L2
6	Explain how to generate properties automatically based on field definitions with examples	CO <sub>5</sub>	L2
7	Explain how to encapsulate logical fields by using properties in c#	CO6	L2
8	What is an indexer? Lists and explain set of operators provided by c# that you can use to access and manipulate the individual bits in an int	CO6	L1
9	Explain how to control read access to indexers by declaring get accessors with example	CO6	L2
10	Explain how to control write access to indexers by declaring set accessors with example	CO6	L2
e	Experiences		
1	Experiences		
2			+

Samuel Control	SKIT	Teaching Process	Rev No.: 1.0
	Doc Code:	SKIT.Ph5b1.F02	Date: 27-02-2019
	Title:	Course Plan	Page: 19 / 27

## E2. CIA EXAM – 2

## a. Model Question Paper - 2

Crs C	Code:	15CS564	Sem:	V	Marks:	30	Time:	75	minute	S	
Cour	se:	DOT NET	FRAMEWC	rK			·	•			
-	-	Note: Ansv	wer any 2 c	juestions, ea	ch carry ed	qual mar	ks.		Marks	CO	Level
1	a	Explain hounded			overflow b	by using	g the checked	and	5	CO2	L2
		Explain how keywords.	w to raise (	exceptions fr	om your o	vn meth	ods using the tl	hrow	3	CO2	L2
		Explain hovestatements		e exceptions	by using th	e try, cat	tch and finally		7	CO2	
					OR						
2	а	Define clas	ss. Explain ł	now class is ι	used in .net.				3	CO2	L3
	b	Briefly exp	Define class. Explain how class is used in .net.  Briefly explain static methods and data.						5	CO2	L3
	С	Describe b	riefly the c	ontrolling ac	ccessibility i	n .net			7	CO2	L3
3	а	Explain the	difference	between a v	value type a	and refer	rence type.		5	CO2	L3
	b	Explain ho	w enumera	tion type is c	declared an	d used ir	n .net.		5	CO2	L3
	С	Explain the	concept c	f boxing and	l unboxing v	with exar	mples		5	CO2	L1
					OR						
4	а	Define null	values and	d nullable typ	oes with exa	amples			8	CO3	L2
	b	Explain ho	w a structu	re type is de	clared and	used in .	net.		7	CO3	L2

Sommer Con	SKIT	Teaching Process	Rev No.: 1.0
	Doc Code:	SKIT.Ph5b1.F02	Date: 27-02-2019
	Title:	Course Plan	Page: 20 / 27

## b. Assignment – 2

Note: A distinct assignment to be assigned to each student.

Cris Code: JOCS664 Sem: V Marks: 5 / 10   Time: 90 - 120 minutes  Course: DOT NET FRAMEWORK  Note: Each student to answer 2-3 assignments. Each assignment carries equal mark.  SNo USN Assignment Description Marks: 6 CO Level  1 Write a method that can accept any number of arguments by using the params keyword.  2 Discus method overloading? Explain with example. 4 CO3 L2  3 Explain how to create a derived class that inherits features from a base class with example.  4 Explain how to create a derived class that inherits features from a base class with example.  5 How to limit accessibility within an inheritance hierarchy by 7 CO3 L8  1 How to limit accessibility within an inheritance hierarchy by 1 CO4 L2  2 Define extension methods as an alternative mechanism to 1 Using inheritance with examples  5 Define an interface specifying the signatures and return types 1 Organism how to implement an interface in a structure or class 1 Organism how to implement an interface in a structure or class 1 Organism how to implement an interface in a structure or class 1 Organism how to implement an interface in a structure or class 1 Organism how to implement an interface in a structure or class 1 Organism how to implement an interface in a structure or class 1 Organism how to implement an interface in a structure or class 1 Organism how to implement an interface in a structure or class 1 Organism how 1 Organis				Model Assignment Questions			
Note: Each student to answer 2-3 assignments. Each assignment carries equal mark.	Crs C	ode:	15CS564	Sem: V Marks: 5 / 10 Time:	90 – 120	minute	S
SNo         USN         Assignment Description         Marks         CO         Level           1         Write a method that can accept any number of arguments by using the params keyword.         6         CO3         L1           2         Discus method overloading? Explain with example.         4         CO3         L2           3         Explain how to create a derived class that inherits features from a base class with example.         6         CO3         L2           4         Explain how to implement an interface in a structure or class with examples.         6         CO3         L2           5         How to limit accessibility within an inheritance hierarchy by using the protected keyword, explain with example.         7         CO3         L6           6         Define extension methods as an alternative mechanism to using inheritance with examples         4         CO4         L2           7         Define an interface specifying the signatures and return types of methods with examples         4         CO4         L2           8         Explain how to implement an interface in a structure or class with examples         4         CO4         L2           9         Explain how to implement an interface in a structure or class with example         5         CO4         L2           10         Write code in c# that runs when an object is destroyed <td>Cours</td> <td>se:</td> <td>DOT NE</td> <td>T FRAMEWORK</td> <td></td> <td></td> <td></td>	Cours	se:	DOT NE	T FRAMEWORK			
Write a method that can accept any number of arguments by dising the params keyword.    Write a method overloading? Explain with example.   4   CO3   L2	Note:	Each	student	to answer 2-3 assignments. Each assignment carries equal m	ark.		
using the params keyword.    2	SNo	l	JSN	Assignment Description	Marks	СО	Level
Discus method overloading? Explain with example.  Explain how to create a derived class that inherits features from a base class with example.  Explain how to implement an interface in a structure or class with examples.  How to limit accessibility within an inheritance hierarchy by using the protected keyword, explain with example.  Define extension methods as an alternative mechanism to using inheritance with examples.  Define an interface specifying the signatures and return types of methods with examples.  Define an interface specifying the signatures and return types of methods with examples.  Explain how to implement an interface in a structure or class with examples.  Explain how to manage system resource by using garbage collection with example.  Write code in c# that runs when an object is destroyed.  Write code in c# that runs when an object is destroyed.  Explain how to control lead access to data in an object with examples.  Explain how to control read access to properties by declaring get accessors with example.  Explain how to control write access to properties by using 4 CO5 L2 get accessors with example.  Explain how to control write access to properties by declaring structures and classes.  Explain how to control write access to properties by declaring set accessors with example.  Explain how to control write access to properties by declaring set accessors with example.  Explain how to create interfaces that declare properties with example.  Explain how to generate properties automatically based on 5 CO5 L2 example.  Explain how to generate properties automatically based on 5 CO5 L2 example.  Explain how to encapsulate logical fields by using properties 6 CO6 L2 in c#  What is an indexer? Lists and explain set of operators provided by c# that you can use to access and manipulate the individual bits in an int individual bits in an int in dividual bits in an int of control read access to indexers by declaring 8 CO6 L2 get accessors with example	1				бу 6	CO3	L1
Explain how to create a derived class that inherits features from a base class with example.  Explain how to implement an interface in a structure or class with examples.  How to limit accessibility within an inheritance hierarchy by using the protected keyword, explain with example.  Define extension methods as an alternative mechanism to using inheritance with examples  Define an interface specifying the signatures and return types of methods with examples  Explain how to implement an interface in a structure or class with examples  Explain how to manage system resource by using garbage collection with example  Write code in c# that runs when an object is destroyed  Write code in c# that runs when an object is destroyed  Write code in c# that runs when an object is destroyed  Explain how to control read access to data in an object with examples  Explain how to control read access to data in an object with examples  Explain how to control read access to properties by declaring get accessors with example  Explain how to control write access to properties by using structures and classes.  Explain how to control write access to properties by declaring structures and classes.  Explain how to control write access to properties by declaring set accessors with example  Explain how to control write access to properties by declaring set accessors with example  Explain how to control write access to properties by declaring set accessors with example  Explain how to ceate interfaces that declare properties with set accessors with example  Explain how to centrol write access to properties of code L2 in c#  What is an indexer? Lists and explain set of operators provided by c# that you can use to access and manipulate the individual bits in an int  Explain how to control write access to indexers by declaring get accessors with example  Explain how to control write access to indexers by declaring set accessors with example	2			Discus method overloading? Explain with example.	4	CO3	L2
Explain how to implement an interface in a structure or class with examples.  How to limit accessibility within an inheritance hierarchy by using the protected keyword, explain with example.  Define extension methods as an alternative mechanism to using inheritance with examples  Define an interface specifying the signatures and return types of methods with examples  Explain how to implement an interface in a structure or class with example sexplain how to implement an interface by using garbage collection with example  Explain how to manage system resource by using garbage collection with example with example sexplain how to controlled access to data in an object with examples.  Explain how to control read access to properties by declaring get accessors with example  Explain how to control read access to properties by using structures and classes.  Explain how to control write access to properties by declaring set accessors with example example sexplain how to create interfaces in c# containing properties by declaring set accessors with example example explain how to create interfaces that declare properties with example explain how to encapsulate logical fields by using properties in explain how to encapsulate logical fields by using properties example explain how to encapsulate logical fields by using example explain how to encapsulate logical fields by using example explain how to control read access to indexers by declaring explain how to control write access to indexers by declaring explain how to control write access to indexers by declaring explain how to control write access to indexers by declaring explain how to con	3			Explain how to create a derived class that inherits feature	es 6	CO3	L2
How to limit accessibility within an inheritance hierarchy by using the protected keyword, explain with example.  Define extension methods as an alternative mechanism to using inheritance with examples  Define an interface specifying the signatures and return types of methods with examples  Explain how to implement an interface in a structure or class with examples  Explain how to manage system resource by using garbage collection with example  Write code in c# that runs when an object is destroyed  Write code in c# that runs when an object is destroyed  Discuss the properties? Explain how to create and use properties to provide controlled access to data in an object with example.  Explain how to control read access to properties by declaring get accessors with example  Explain how to control write access to properties by using structures and classes.  Explain how to control write access to properties by declaring set accessors with example  Explain how to create interfaces that declare properties with set accessors with example  Explain how to create interfaces that declare properties with set accessors with example  Explain how to generate properties automatically based on field definitions with examples  Explain how to encapsulate logical fields by using properties for confield definitions with examples  Explain how to encapsulate logical fields by using properties for confield definitions with examples  What is an indexer? Lists and explain set of operators provided by c# that you can use to access and manipulate the individual bits in an int  What is an indexer? Lists and explain set of operators provided by c# that you can use to access and manipulate the individual bits in an int  Explain how to control read access to indexers by declaring get accessors with example  Explain how to control write access to indexers by declaring set accessors with example	4			Explain how to implement an interface in a structure or class	s 6	CO3	L2
Define extension methods as an alternative mechanism to using inheritance with examples Define an interface specifying the signatures and return types of methods with examples Explain how to implement an interface in a structure or class with examples Explain how to manage system resource by using garbage collection with example  Write code in c# that runs when an object is destroyed Discuss the properties? Explain how to create and use properties to provide controlled access to data in an object with examples.  Explain how to control read access to properties by declaring get accessors with example  Explain how to control write access to properties by using structures and classes.  Explain how to control write access to properties by declaring structures and classes.  Explain how to control write access to properties by declaring structures and classes.  Explain how to control write access to properties by declaring structures and classes.  Explain how to create interfaces that declare properties with Explain how to create interfaces that declare properties with Explain how to generate properties automatically based on stelled definitions with examples  Explain how to generate properties automatically based on stelled definitions with examples  Explain how to encapsulate logical fields by using properties success in c#  What is an indexer? Lists and explain set of operators provided by c# that you can use to access and manipulate the individual bits in an int  Explain how to control read access to indexers by declaring accessors with example  Explain how to control read access to indexers by declaring Explain how to control read access to indexers by declaring accessors with example	5			How to limit accessibility within an inheritance hierarchy b	у 7	CO3	L6
Define an interface specifying the signatures and return types of methods with examples  Explain how to implement an interface in a structure or class with examples  Explain how to manage system resource by using garbage collection with example  Explain how to manage system resource by using garbage collection with example  Write code in c# that runs when an object is destroyed 5 CO4 L6  Discuss the properties? Explain how to create and use properties to provide controlled access to data in an object with examples.  Explain how to control read access to properties by declaring get accessors with example  Demonstrate interfaces in c# containing properties by using structures and classes.  Explain how to control write access to properties by declaring set accessors with example  Explain how to create interfaces that declare properties with example explain how to create interfaces that declare properties with example  Explain how to generate properties automatically based on field definitions with examples  Explain how to encapsulate logical fields by using properties Explain how to encapsulate logical fields by using properties with provided by c# that you can use to access and manipulate the inclividual bits in an int  What is an indexer? Lists and explain set of operators provided by c# that you can use to access and manipulate the inclividual bits in an int  Explain how to control read access to indexers by declaring get accessors with example  Explain how to control write access to indexers by declaring get accessors with example	6			Define extension methods as an alternative mechanism t	0 4	CO <sub>4</sub>	L2
Explain how to implement an interface in a structure or class with examples  Explain how to manage system resource by using garbage collection with example  Write code in c# that runs when an object is destroyed 5 CO4 L6  Discuss the properties? Explain how to create and use properties to provide controlled access to data in an object with examples.  Explain how to control read access to properties by declaring get accessors with example  Demonstrate interfaces in c# containing properties by using structures and classes.  Explain how to control write access to properties by declaring structures and classes.  Explain how to control write access to properties by declaring structures and classes.  Explain how to control write access to properties by declaring structures and classes.  Explain how to create interfaces that declare properties with Explain how to create interfaces that declare properties with sexample Explain how to generate properties automatically based on field definitions with examples  Explain how to encapsulate logical fields by using properties Explain how to encapsulate logical fields by using properties and manipulate the individual bits in an int  What is an indexer? Lists and explain set of operators provided by c# that you can use to access and manipulate the individual bits in an int  Explain how to control read access to indexers by declaring aget accessors with example Explain how to control write access to indexers by declaring Explain how to control write access to indexers by declaring Explain how to control write access to indexers by declaring Explain how to control write access to indexers by declaring Explain how to control write access to indexers by declaring Explain how to control write access to indexers by declaring Explain how to control write access to indexers by declaring Explain how to control write access to indexers by declaring Explain how to control write access to indexers by declaring Explain how to control write access to indexers by declaring Explain how to the	7			Define an interface specifying the signatures and return type	es 4	CO <sub>4</sub>	L2
Explain how to manage system resource by using garbage collection with example  Write code in c# that runs when an object is destroyed  Discuss the properties? Explain how to create and use properties to provide controlled access to data in an object with examples.  Explain how to control read access to properties by declaring get accessors with example  Demonstrate interfaces in c# containing properties by using structures and classes.  Explain how to control write access to properties by declaring set accessors with example  Explain how to control write access to properties with example  Explain how to create interfaces that declare properties with example  Explain how to generate properties automatically based on field definitions with examples  Explain how to encapsulate logical fields by using properties field definitions with examples  What is an indexer? Lists and explain set of operators provided by c# that you can use to access and manipulate the individual bits in an int  What is an indexer access to indexers by declaring explain how to control read access to indexers by declaring explain how to control write access to indexers by declaring explain how to control write access to indexers by declaring explain how to control write access to indexers by declaring explain how to control write access to indexers by declaring explain how to control write access to indexers by declaring explain how to control write access to indexers by declaring explain how to control write access to indexers by declaring explain how to control write access to indexers by declaring explain how to control write access to indexers by declaring explain how to control write access to indexers by declaring explain how to control write access to indexers by declaring explain how to control write access to indexers by declaring explain how to control write access to indexers by declaring explain how to control write access to indexers by declaring explain how to control write access to indexers by declaring explain how to control	8			Explain how to implement an interface in a structure or class	SS 4	CO <sub>4</sub>	L2
Write code in c# that runs when an object is destroyed  Discuss the properties? Explain how to create and use properties to provide controlled access to data in an object with examples.  Explain how to control read access to properties by declaring get accessors with example  Demonstrate interfaces in c# containing properties by using structures and classes.  Explain how to control write access to properties by declaring set accessors with example  Explain how to control write access to properties by declaring set accessors with example  Explain how to create interfaces that declare properties with set accessing example  Explain how to generate properties automatically based on set accessing field definitions with examples  Explain how to encapsulate logical fields by using properties set in c#  What is an indexer? Lists and explain set of operators set individual bits in an int set of operators access and manipulate the individual bits in an int set of operators set individual bits in an int set of operators set of provided by c# that you can use to access and manipulate the individual bits in an int set of operators set of operators set of operators of the individual bits in an int set of operators set of operators set of operators of the individual bits in an int set of operators of the individual bits in an int set of operators of the individual bits in an int set of operators of the individual bits in an int set of operators of the individual bits in an int set of operators of the individual bits in an int set of operators of the individual bits in an int set of operators of the individual bits in an int set of operators of the individual bits in an int set of operators of the individual bits in an int set of operators of the individual bits in an int set of operators of the individual bits in an int set of operators of the individual bits in an int set of operators of the individual bits in an int set of operators of the individual bits in an int set of operators of the individual bits in an int set of op	9			Explain how to manage system resource by using garbag	e 4	CO <sub>4</sub>	L2
Discuss the properties? Explain how to create and use properties to provide controlled access to data in an object with examples.  Explain how to control read access to properties by declaring get accessors with example  Demonstrate interfaces in c# containing properties by using structures and classes.  Explain how to control write access to properties by declaring set accessors with example  Explain how to control write access to properties by declaring set accessors with example  Explain how to create interfaces that declare properties with example  Explain how to generate properties automatically based on set accessing the definitions with examples  Explain how to encapsulate logical fields by using properties successing the inception of the provided by c# that you can use to access and manipulate the individual bits in an int  What is an indexer? Lists and explain set of operators provided by c# that you can use to access and manipulate the individual bits in an int  Explain how to control read access to indexers by declaring explain how to control read access to indexers by declaring explain how to control write access to indexers by declaring explain how to control write access to indexers by declaring explain how to control write access to indexers by declaring explain how to control write access to indexers by declaring explain how to control write access to indexers by declaring explain how to control write access to indexers by declaring explain how to control write access to indexers by declaring explain how to control write access to indexers by declaring explain how to control write access to indexers by declaring explain how to control write access to indexers by declaring explain explain how to control write access to indexers by declaring explain how to control write access to indexers by declaring explain how to control write access to indexers by declaring explain how to control write access to indexers by declaring explain how to control write access to indexers by declaring explain how to c	10			'	5	CO <sub>4</sub>	L6
Explain how to control read access to properties by declaring get accessors with example  Demonstrate interfaces in c# containing properties by using structures and classes.  Explain how to control write access to properties by declaring set accessors with example  Explain how to create interfaces that declare properties with example  Explain how to generate properties automatically based on field definitions with examples  Explain how to encapsulate logical fields by using properties for containing properties and provided by c# that you can use to access and manipulate the individual bits in an int  Explain how to control read access to indexers by declaring get accessors with example  Explain how to control write access to indexers by declaring set accessors with example  Explain how to control write access to indexers by declaring set accessors with example  Explain how to control write access to indexers by declaring set accessors with example				Discuss the properties? Explain how to create and us properties to provide controlled access to data in an object	e 6		
Demonstrate interfaces in c# containing properties by using structures and classes.  Explain how to control write access to properties by declaring set accessors with example  Explain how to create interfaces that declare properties with example  Explain how to generate properties automatically based on field definitions with examples  Explain how to encapsulate logical fields by using properties for containing properties in c#  What is an indexer? Lists and explain set of operators provided by c# that you can use to access and manipulate the individual bits in an int  We have to encapsulate logical fields by using properties for containing properties for containing properties automatically based on field definitions with example for containing properties for containing properties automatically based on field definitions with example for containing properties for	12		Explain how to control read access to properties by declaring			CO <sub>5</sub>	L2
set accessors with example  Explain how to create interfaces that declare properties with example  Explain how to generate properties automatically based on field definitions with examples  Explain how to encapsulate logical fields by using properties for c#  What is an indexer? Lists and explain set of operators provided by c# that you can use to access and manipulate the individual bits in an int  Explain how to control read access to indexers by declaring get accessors with example  Explain how to control write access to indexers by declaring set accessors with example	13			Demonstrate interfaces in c# containing properties by usin	g 4	CO <sub>5</sub>	L3
example  Explain how to generate properties automatically based on field definitions with examples  Explain how to encapsulate logical fields by using properties for comparison of comparison of comparison of field definitions with examples  Explain how to encapsulate logical fields by using properties for comparison of com	14				g 5	CO <sub>5</sub>	L2
field definitions with examples  Explain how to encapsulate logical fields by using properties 6 CO6 L2 in c#  What is an indexer? Lists and explain set of operators provided by c# that you can use to access and manipulate the individual bits in an int  Explain how to control read access to indexers by declaring get accessors with example  Explain how to control write access to indexers by declaring 8 CO6 L2 set accessors with example	15				:h 8	CO <sub>5</sub>	L2
in c#  What is an indexer? Lists and explain set of operators provided by c# that you can use to access and manipulate the individual bits in an int  Explain how to control read access to indexers by declaring get accessors with example  Explain how to control write access to indexers by declaring set accessors with example  Explain how to control write access to indexers by declaring set accessors with example	16			, ,	on 5	CO <sub>5</sub>	L2
provided by c# that you can use to access and manipulate the individual bits in an int  Explain how to control read access to indexers by declaring get accessors with example  Explain how to control write access to indexers by declaring set accessors with example  Explain how to control write access to indexers by declaring set accessors with example	17				es 6	CO6	L2
get accessors with example  Explain how to control write access to indexers by declaring 8 CO6 L2 set accessors with example	18			provided by c# that you can use to access and manipulat the individual bits in an int	е	CO6	L1
Explain how to control write access to indexers by declaring 8 CO6 L2 set accessors with example	19				g 8	CO6	L2
21	20			Explain how to control write access to indexers by declaring	g 8	CO6	L2
	21						

Souther Co.	SKIT	Teaching Process	Rev No.: 1.0
	Doc Code:	SKIT.Ph5b1.F02	Date: 27-02-2019
	Title:	Course Plan	Page: 21 / 27

# D<sub>3</sub>. TEACHING PLAN - 3

# Module - 5

Title:	Collections	Appr Time:	8 Hrs
а	Course Outcomes	-	Blooms
-	The student should be able to:	-	Level
1	Develop own operator with behavior	CO7	L3
2	Design queries to query in-memory data	CO8	L6
b	Course Schedule		
Class No	Module Content Covered	СО	Level
1	Enumerating Collections	CO7	L3
2	Decoupling application logic and handling events	CO7	L3
3	Decoupling application logic and handling events	CO7	L3
4	Querying in-memory data by using query expressions	CO8	L6
5	Querying in-memory data by using query expressions	CO8	L6
6	Querying in-memory data by using query expressions	CO8	L6
7	Operator overloading	CO7	L3
8	Operator overloading	CO7	L3
С	Application Areas	CO	Level
1	Web based services	CO7	L4
2	Collections for data customization	CO8	L6
d	Review Questions		
1	What exactly is an enumerable collection? Demonstrate an enumerator that you can be used to iterate over the elements in a collection with examples	CO7	L3
2	Implement an enumerator automatically by creating an iterator in C#. Explain.	CO7	L4
3	Explain how to provide additional iterators that can step through the elements of a collection in different sequences.	CO7	L2
4	Implementing the IEnumerable interface in c#	CO7	L4
5	Implement how to call a method through a delegate in c#.	CO7	L4
6	Explain how to create an instance of a delegate to refer to a specific method	CO7	L6
7	Implement how to call a method through a delegate in c#	CO8	L4
8	Define a lambda expression to specify the code to be executed by a delegate	CO8	L2
9	Explain how Declare an event field with examples	CO8	L2
10	Explain how to handle an event by using a delegate	CO8	L2
11			
е	Experiences	-	-
1			
2			

Sommer Con	SKIT	Teaching Process	Rev No.: 1.0
	Doc Code:	SKIT.Ph5b1.F02	Date: 27-02-2019
	Title:	Course Plan	Page: 22 / 27

### E3. CIA EXAM – 3

### a. Model Question Paper - 3

Crs C	Code:	15CS564	Sem:	V	Marks:	30	Time:	75	minute	minutes			
Cour	se:	DOT NET F	RAMEWOR	K									
-	-	Note: Answ	er any 2 qu	estions, ead	ch carry eq	ual marks	5.		Marks	CO	Level		
1	a	Explain how	v to create i	nterfaces th	at declare ¡	oroperties	with example		8	CO5	L2		
			plain how to generate properties automatically based on fiden initions with examples								L2		
			-		OR								
2				le additiona n in differen			step through	the	7	CO7	L2		
	b	Implementi	ng the IEnu	merable inte	erface in c#	ŧ			8	CO7	L4		
3				ts and expla and manipu			rovided by c# t s in an int	that	7	CO6	L1		
			w to conti with exampl		ccess to i	ndexers l	oy declaring	get	8	CO6	L2		
					OR								
4	а	Explain how	v Declare ar	event field	with exam	ples			8	CO8	L2		
	b	Explain how	v to handle a	an event by	using a del	.egate			7	CO8	L2		

## b. Assignment – 3

Note: A distinct assignment to be assigned to each student.

	7 ( 0.10		giirient to b		odel Assigi			ns				
Crs C	ode:	15CS564	ı Sem:	V	Mark		5 / 10		ime:	90 – 120	minute	 S
Cours			T FRAMEW	ORK	I			-				
Note:	Each	student	to answer 2-	3 assigr	nments. Ea	ch assi	gnment	carrie	s equal m	ark.		
SNo	Į	JSN			Assignmen	t Desc	ription		•	Marks	СО	Level
1			What exact enumerator in a collection	that yo	u can be u						CO7	L3
2			Implement iterator in C			autom	natically	by o	creating a	an	CO7	L4
3			Explain how through the							ep	CO7	L2
4			Implementi	ng the IE	Enumerabl	e inter	face in c	:#			CO7	L4
5			Implement	how to a	call a meth	od thro	ough a c	delega	te in c#.		CO7	L4
6			Explain how specific met		ate an inst	ance c	of a dele	egate t	o refer to	a	CO7	L6
7			Implement	how to a	call a meth	od thro	ough a c	delega	te in c#		CO8	L4
8			Define a la executed by			n to s	specify	the c	ode to k	pe	CO8	L2
9			Explain how	Declar	e an event	field w	ith exan	nples			CO8	L2
10			Explain how	to hand	dle an ever	nt by u	sing a de	elegat	е		CO8	L2
11			Explain how	Declar	e an event	field w	ith exan	nples.				

Some	SKIT	Teaching Process	Rev No.: 1.0
	Doc Code:	SKIT.Ph5b1.F02	Date: 27-02-2019
	Title:	Course Plan	Page: 23 / 27

Copyright ©2017. cAAS. All rights reserved.

## F. EXAM PREPARATION

1. University Model Question Paper

The same of the sa	SKIT	Teaching Process	Rev No.: 1.0
	Doc Code:	SKIT.Ph5b1.F02	Date: 27-02-2019
11500	Title:	Course Plan	Page: 24 / 27

Copyright ©2017. cAAS. All rights reserved. DOT NET FRAMEWORK Month / Year Course: V Marks: 80 Crs Code: 15CS564 Sem: Time: 180 minutes **Note** Answer all FIVE full questions. Marks CO Level What is Namespace? How namespace is used to solve name-clashing 80 CO<sub>1</sub> L2 problem, Explainwith Example. What is an assembly? Explain with Examples CO<sub>1</sub> L3 Create a C# console application 07 CO1 L3 Explain how the precedence and associativity of operators determine CO<sub>1</sub> 80 L2 2 and howexpressions are evaluated in C#. Describe Method overloading? Explain useful of method overloading with 80 CO<sub>1</sub> b L3 example Explain how to use while, for, and do statements to execute code CO1 L2 04 repeatedly while someBoolean condition is true with example. What is constructor? Explain necessity of overloading constructors with C<sub>02</sub> L2 3 example. Explain the differences between a value type and a reference type with CO<sub>2</sub> b 06 L3 example Explain the purpose of encapsulation with example CO2 06 L2 С OR What is boxing and unboxing? Explain with examples. CO2 06 L2 4 Explain how to control the accessibility of members by using the public CO<sub>2</sub> 80 L3 and private keywords with examples. Explain how to Declare, Create and use an enumeration type with CO<sub>2</sub> 06 L2 Examples. Write a method that can accept any number of arguments by using the 80 CO3 5 L3 params keyword. What is method overloading? Explain with example. CO3 L2 b 07 Explain how to create a derived class that inherits features from a base 05 CO<sub>3</sub> L2 class withexample. OR Explain how to implement an interface in a structure or class with 6 80 CO<sub>4</sub> L2 examples. b Explain how to manage system resources by using garbage collection 06 CO<sub>4</sub> L3 with example Write code in c# that runs when an object is destroyed. CO4 06 L2 What are properties? Explain how to create and use properties to provide 08 CO5 12 7 controlled accessto data in an object with examples. Explain how to create automatic properties and how to use properties CO5 L2 05 when initializingobjects. Explain how to control read access to properties by declaring get CO5 L2 07 accessors with example. OR Explain how to control read access to indexers by declaring get CO6 8 L2 05 accessors with example. Explain the differences between indexers and arrays with examples. CO6 b 07 L2 What are the problem with the object type? Explain the purpose of 80 CO6 L2 generics with examples. What exactly is an enumerable collection? Explain how to manually 80 CO7 L2 9 define anenumerator that you can use to iterate over the elements in a collection with examples. Implement an enumerator automatically by creating an iterator in C#. CO7 L4 Explain.

18	THE PARTY OF THE P	SKIT	Teaching Process	Rev N	0.: 1.0			
		Doc Code:	SKIT.Ph5b1.F02	Date:	27-02-2	2019		
160	ALCON		Course Plan	Page:	25 / 27	7		
Copyrig	jht ©2017. (	CAAS. All rights reserved						
	С	Implementing	the IEnumerable interface in c#.	05	CO7	L4		
			OR					
10		Explain how t method.	o create an instance of a delegate to refer to a specific	07	CO8	L3		
	р	Implement ho	w to call a method through a delegate in c#.	05	CO8	L2		
	С	Explain how to	handle an event by using a delegate.	08	CO8	L2		

	SKIT	Teaching Process	Rev No.: 1.0
	Doc Code:	SKIT.Ph5b1.F02	Date: 27-02-2019
	Title:	Course Plan	Page: 26 / 27

## 2. SEE Important Questions

Cour		DOT NET FRAMEWORK Month.	/ Vear		
	Code:	15CS564 Sem: V Crs Code: 15CS564 Sem:	real	V	
C13 (		Answer all FIVE full questions. All questions carry equal marks.	_	· -	
Мо		This wor all TIVE rail quosilons. The questions early equal marks.	Marks	СО	Year
dule					
1	1	What is Namespace? How namespace is used to solve name-clashing	10	CO1	2016
		problem, Explain with Example.			
	2	What is an assembly? Explain with Examples	06	CO1	2016
	3	Explain how to create a blank universal windows platform app using	04	CO1	2016
	1	visual studio 2015 with example.  4 Create a C# console application			2016
	<u>4</u> 5	Explain how the precedence and associativity of operators determine	09 10	CO <sub>1</sub>	2010
	3	and how expressions are evaluated in C#.			
2	1	What is constructor? Explain necessity of overloading constructors with example.	4	CO2	2017
	2	Explain how arguments are passed as method parameters by using the ref and out keywords		CO2	2017
	3	Explain the differences between a value type and a reference type with example	10	CO2	2017
	4	Write and call your own constructors in C#. Explain how to create anonymous classes with examples.	05	CO2	2016
	5	Explain how to control the accessibility of members by using the public and privatekeywords with examples.	06	CO2	2015
3	1	Write a method that can accept any number of arguments by using the	08	CO3	
5	_	params keyword.		005	
	2	What is method overloading? Explain with example.	05	CO <sub>3</sub>	
	3	Explain how to create a derived class that inherits features from a base class with example.	07	CO3	2016
	4	Explain how to implement an interface in a structure or class with examples.	5	CO4	
	5	Write and call your own constructors in C#. Explain how to create anonymous classes with examples.	5	CO <sub>4</sub>	
4	1	Write a method that can accept any number of arguments by using the params keyword.	08	CO <sub>5</sub>	
	2	Explain the differences between methods that take parameter arrays and methods thattake optional parameters	05	CO6	
	3	What is method overloading? Explain with example.	07	CO5	
	4	Explain how to create a derived class that inherits features from a base class with example	08	CO6	
	5	.Write and call your own constructors in C#. Explain how to create anonymous classes with examples.	07	CO6	
5	1	Implement an enumerator automatically by creating an iterator in C#. Explain	08	CO7	2016
	2	Explain how to provide additional iterators that can step through the elements of a collection in different sequences.	07	CO7	2010
	3	Implementing the IEnumerable interface in c#.	05	CO8	
	4	Implement how to call a method through a delegate in c#.	07	CO8	2017
	5	Explain how to create an instance of a delegate to refer to a specific method.	05	CO8	2015

	SKIT	Teaching Process	Rev No.: 1.0
	Doc Code:	SKIT.Ph5b1.F02	Date: 27-02-2019
	Title:	Course Plan	Page: 27 / 27