Maharashtra State Board of Technical Education, Mumbai
Teaching and Examination Scheme for Post S.S.C. Diploma Courses

Program Name : Computer Engineering Groups

Program Code : CO/CM/CW	With Effect From Academic Year: 2017 - 18
Duration of Program : 6 Semesters	Duration : 16 Weeks
Semester : Fourth	Scheme : I

S. N.	Course Title	Course Abbreviation	Course Code	Teaching Scheme			Credit (L+T+P)	Examination Scheme														Grand Total
									Theory							Practical						
				L	T	P		Exam Duration in Hrs.	ESE		PA		Total		ESE		PA		Total		
									Max Marks	Min Marks	Max Marks	Min Marks	Max Marks	Min Marks	Max Marks	Min Marks	Max Marks	Min Marks	Max Marks	Min Marks	
1	Java Programming	JPR	22412	3	-	4	7	3	70	28	30*	00	100	40	50#	20	50	20	100	40	200
2	Software Engineering	SEN	22413	3	-	2	5	3	70	28	30*	00	100	40	25@	10	25	10	50	20	150
3	Data Communication and Computer Network	DCC	22414	4	-	2	6	3	70	28	30*	00	100	40	25@	10	25	10	50	20	150
4	Microprocessors	MIC	22415	4	-	2	6	3	70	28	30*	00	100	40	25@	10	25	10	50	20	150
5	GUI Application Development using VB.Net	GAD	22034	2	-	4	6	--	--	--	--	--	--	--	50#	20	50~	20	100	40	100
	Total			16	-	14	30	--	280	--	120	--	400	--	175	--	175	--	350	--	750

Student Contact Hours Per Week: **30 Hrs.**   Medium of Instruction: **English**

Theory and practical periods of 60 minutes each.   Total Marks : **750**

Abbreviations: ESE- End Semester Exam, PA- Progressive Assessment, L - Lectures, T - Tutorial, P - Practical

@ Internal Assessment, # External Assessment, *# On Line Examination, ^ Computer Based Assessment

* Under the theory PA, Out of 30 marks, 10 marks are for micro-project assessment to facilitate integration of COs and the remaining 20 marks is the average of 2 tests to be taken during the semester for the assessment of the cognitive domain LOs required for the attainment of the COs.

~ For the courses having ONLY Practical Examination, the PA marks Practical Part - with 60% weightage and Micro-Project Part with 40% weightage

➢ **If Candidate not securing minimum marks for passing in the "PA" part of practical of any course of any semester then the candidate shall be declared as "Detained" for that semester.**

➢ **In-Plant Training during Summer vacation for minimum Six Weeks at the end of Fourth Semester (Second Year).**

Program Name	: Computer Engineering Program Group
Program Code	: CO/CM/IF/CW
Semester	: Fourth
Course Title	: GUI Application Development using VB.Net
Course Code	: 22034

1.   RATIONALE

VB.NET is the programming language based on Object Oriented Concepts which is prominently used to develop GUI based Applications. Graphical User Interface (GUI) based application includes various user friendly controls to accept or display data. This course will give the students an in-depth understanding of the concepts used in VB .NET and necessary skills to use programming techniques to develop .NET based applications and deploy the same.

2.   COMPETENCY

The aim of this course is to help the student to attain the following industry identified competency through various teaching learning experiences:
- **Develop GUI based application using VB.net.**

3.   COURSE OUTCOMES (COs)

The theory, practical experiences and relevant soft skills associated with this course are to be taught and implemented, so that the student demonstrates the following *industry oriented* COs associated with the above mentioned competency:
- a. Use Visual Studio IDE to design application.
- b. Develop GUI Application using Form Controls and its events.
- c. Apply Object Oriented concepts in GUI Application.
- d. Use Data access controls to store data in Database and retrieve it.
- e. Use Data Binding in GUI Application.

4.   TEACHING AND EXAMINATION SCHEME

Teaching Scheme			Credit (L+T+P)	Examination Scheme												
					Theory						Practical					
L	T	P		Paper Hrs.	ESE		PA		Total		ESE		PA		Total	
					Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min
2	-	4	6	--	--	--	--	--	--	--	50#	20	50~	20	100	40

'#':No practical Examination, *(~²): For the **practical only courses,** the PA has two components under practical marks i.e. the assessment of practicals (seen in section 6) has a weightage of 60% (i.e. 30 marks) and micro-project assessment (seen in section 12) has a weightage of 40% (i.e. 20 marks). This is designed to facilitate attainment of COs holistically, as there is no theory ESE.*

Legends: L-Lecture; **T** – Tutorial/Teacher Guided Theory Practice; **P** - Practical; **C** – Credit, **ESE** - End Semester Examination; **PA** - Progressive Assessment, **'#':** No Theory Examination

5.   COURSE MAP (with sample COs, PrOs, UOs, ADOs and topics)

This course map illustrates an overview of the flow and linkages of the topics at various levels of outcomes (details in subsequent sections) to be attained by the student by the end of the course, in all domains of learning in terms of the industry/employer identified competency depicted at the centre of this map.
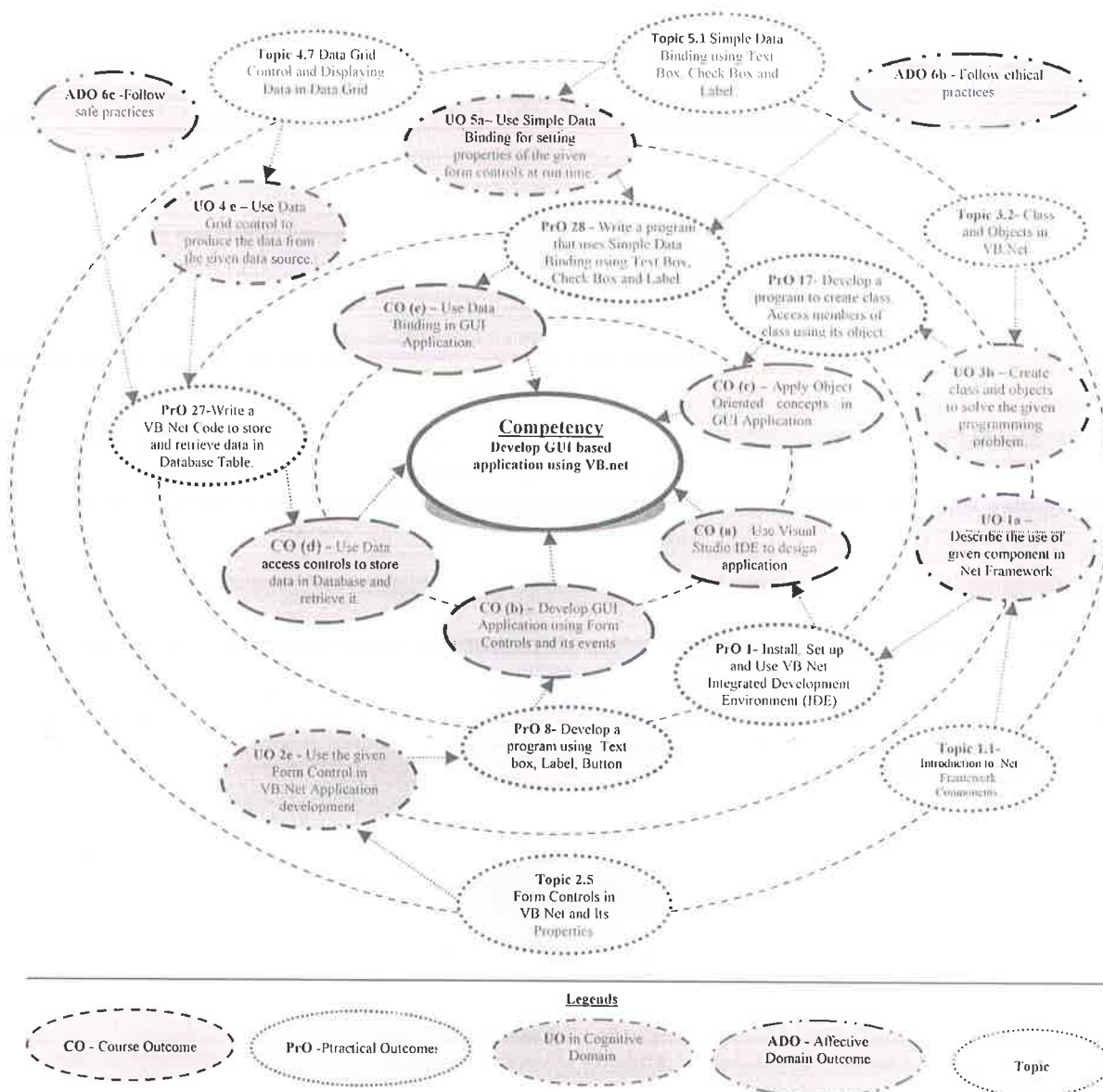

Figure 1 - Course Map

6. SUGGESTED PRACTICALS/ EXERCISES

The practicals in this section are PrOs (i.e. sub-components of the COs) to be developed and assessed in the student for the attainment of the competency.

S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. Required
1.	Install, Set up and Use VB.Net IDE (Integrated Development Environment).	I	
2.	Use Existing Namespaces and Create user defined Namespace in VB.Net.	I	

S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. Required
3.	(a) Write a simple program to display a welcome message using *msgbox()*. (b) Develop programs to solve Arithmetic expressions.	II	02*
4.	Develop programs to demonstrate use of *IF, IF-else* Control structures in VB.net.	II	02*
5.	Develop programs to demonstrate use of *Case* Control structures in VB.net.	II	02*
6.	Develop programs to demonstrate use of *While, DO Loops* in VB.net.	II	02*
7.	Develop programs to demonstrate use of *For, For-each* Loops in VB.net.	II	02*
8.	Develop a program using Text box, Label, Button	II	02*
9.	Develop a program using Radio button, check box,	II	02*
10.	Develop a program using List box, Combo box.	II	02*
11.	Write a program using Picture Box, Panel.	II	02*
12.	Write a program using Tab Control, and Timer.	II	02*
13.	Write a program to perform validation using regular expression and error provider.	II	02*
14.	Write a program to perform validation using regular expression and error provider.	II	02*
15.	Write a program to demonstrate use of Sub-procedures and Parameterized Sub-Procedures.	III	02
16.	Write a program to demonstrate use of Simple function and parameterized Functions.	III	02*
17.	Develop a program to create class. Access members of class using its object.	III	02*
18.	Create constructor to initialize object of class. Use Destructor to de-allocate memory using *finalize* method.	III	02*
19.	Develop a program to inherit members of super class in sub class using simple inheritance.	III	02*
20.	Develop a program to demonstrate Overloading a method	III	02*
21.	Develop a program to demonstrate Overriding in inheritance	III	02*
22.	Develop a program to demonstrate Shadowing in inheritance		02
23.	Construct a program to handle runtime errors by using Exception handling.	III	02*
24.	Write a program to fetch data from table and display in Data Grid.	IV	02*
25.	Write a program to perform following operation using Data Adapter: Fill and Update data in Database.	IV	02*
26.	Write a program to perform following operation using Data Adapter: Fetch data from multiple tables in Dataset.	IV	02
27.	Write a VB.Net Code to store and retrieve data in Database Table.	IV	02*
28.	Write a program that uses Simple Data Binding using Text Box, Check Box and Label.	V	02
29.	Write a program that uses Complex Data Binding using Combo	V	02

S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. Required
	Box.		
30.	Write a program that uses Complex Data Binding using List Box.	V	02
31.	Write a program to Navigate across existing data in table.	V	02
32.	Create Executable file of VB.Net Application and Deploy it to other computer.	V	02*
	Total		**64**

Note

i. *A suggestive list of PrOs is given in the above table. More such PrOs can be added to attain the COs and competency. A judicial mix of minimum 24 or more practical need to be performed, out of which, the practicals marked as '*' are compulsory, so that the student reaches the 'Application Level' of Bloom's Taxonomy' as generally required by the industry.*

ii. *The 'Process' and 'Product' related skills associated with each PrO are to be assessed according to a suggested sample given below:*

S.No.	Performance Indicators	Weightage in %
1	Use of relevant VB.Net tool to design given GUI application.	20
2	Write appropriate code to generate desired output in GUI Application.	30
3	Debug, test and execute the programs/modules.	30
4	Able to answer oral questions.	10
5	Submission of report in time.	10
	Total	**100**

The above PrOs also comprise of the following social skills/attitudes which are Affective Domain Outcomes (ADOs) that are best developed through the laboratory/field based experiences:

    a. Follow safety practices.
    b. Practice good housekeeping.
    c. Demonstrate working as a leader/a team member.
    d. Follow ethical practices.

The ADOs are not specific to any one PrO, but are embedded in many PrOs. Hence, the acquisition of the ADOs takes place gradually in the student when s/he undertakes a series of practical experiences over a period of time. Moreover, the level of achievement of the ADOs according to Krathwohl's 'Affective Domain Taxonomy' should gradually increase as planned below:

    • 'Valuing Level' in 1st year
    • 'Organising Level' in 2nd year and
    • 'Characterising Level' in 3rd year.

7.   MAJOR EQUIPMENT/ INSTRUMENTS REQUIRED

The major equipment with broad specification mentioned here will usher in uniformity in conduct of experiments, as well as aid to procure equipment by authorities concerned.

S. No.	Equipment Name with Broad Specifications	PrO S. No.

S. No.	Equipment Name with Broad Specifications	PrO S. No.
1	Personal computer, (preferably i3-i5 processor based), RAM minimum 2 GB, Hard disk 10 GB minimum available space.	For all Experiments
2	Operating system: Windows 7/8/10	
3	Microsoft Visual Studio 2012 or later.	

8. UNDERPINNING THEORY COMPONENTS

The following topics are to be taught and assessed in order to develop the sample UOs given below for achieving the COs to attain the identified competency. More UOs could be added.

Unit	Major Learning Outcomes (in cognitive domain)	Topics and Sub-topics
Unit – I Overview of GUI Program ming	1a. Describe use of the given component in .Net Framework. 1b. Describe use of use the given element in VB.Net IDE. 1c. Apply the given System Namespace in VB.net Application. 1d. Create Event Handler to respond to the given event.	1.1 Introduction to .Net Framework Components. i. Common Language Runtime (CLR) ii. Microsoft Intermediate Language (MSIL) iii. Just-In-Time Compiler 1.2 Exploring VB.Net IDE 1.3 System Namespaces in VB.Net 1.4 Events and Event handling

Unit– II Decision control and Loop control	2a. Select relevant data type for the given problem with justification.
	2b. Write expression using operators for the given VB.Net Application.
	2c. Use relevant control structure to apply the given criteria for decision making and branching.
	2d. Implement relevant type of loop to solve the given iterative problem.
	2e. Use the given Form Control in VB.Net Application development.
		2.1 Data Types in VB.Net
		2.2 **Operators in VB. Net**
		    ii. Arithmetic Operators
		    iii. Logical Operators
		    iv. Bit Shift Operators
		    v. Relational Operators
		    vi. Assignment Operators
		2.3 **Control Structures**
		    ii. IF Statement
		    iii. IF – ELSE Statement
		    iv. Select Case Statement
		2.4 **Loops in VB.Net**
		    i. For Loop
		    ii. While Loop
		    iii. Do Loop
		    iv. For Each Loop
		2.5 **Form Controls in VB.Net and Its Properties** - Button, Text box, Label, Radio button, Check Box, List Box, Combo Box, Picture Box, Panel, Tab Control, Timer.

Unit– III Object Oriented	3a. Write Sub-procedure/function to solve the given problem. 3b. Create class and objects to solve	3.1 Sub Procedures and Functions 3.2 Class and Objects in VB.Net 3.3 Constructors and Destructors in

Unit	Major Learning Outcomes (in cognitive domain)	Topics and Sub-topics
Programming in VB.Net	the given programming problem. 3c. Create constructor to initialize the given object. 3d. Apply Inheritance to inherit members of the super class in the given problem. 3e. Use Overloading/Overriding/ Shadowing in the given situation. 3f. Develop exception handling mechanism to handle the given exception.	VB.Net 3.4 Inheritance in VB.Net, Simple Inheritance using *Override* Keyword 3.5 Overloading, Overriding and Shadowing 3.6 Exception Handling
Unit-IV Data access in VB.NET	4a. Use the relevant component of ADO.Net architecture based on the given situation. 4b. Select relevant data provider to solve the given problem. 4c. Use the given data provider to the given Access database. 4d. Produce data using Data Adapter Control for communication between the given dataset and the data source. 4e. Use Data Grid control to produce the data from the given data source.	4.1 Architecture of ADO.Net. 4.2 Accessing Data with Server Explorer. 4.3 Data Providers. 4.4 Connections, Data Reader, Data Adapters and Datasets. 4.5 Creating new Data Connection. 4.6 Creating Dataset. 4.7 Data Grid Control and Displaying Data in Data Grid. 4.8 Data Access using Data Adapter.
Unit –V Data Binding and Deployment	5a. Use Simple Data binding for setting properties of the given form control at run time. 5b. Use Complex Data binding for setting properties of the given form control at run time. 5c. Implement VB.Net Application to navigate data in the given database. 5d. Deploy the given VB.Net Application.	5.1 Simple Data Binding using Text Box, Check Box and Label. 5.2 Complex Data Binding using Combo box and List box. 5.3 Navigating Database. 5.4 Deploying VB.Net Application.

Note: To attain the COs and competency, above listed UOs need to be undertaken to achieve the 'Application Level' of Bloom's 'Cognitive Domain Taxonomy'.

9.   SUGGESTED SPECIFICATION TABLE FOR QUESTION PAPER
    –Not Applicable-

10.   SUGGESTED STUDENT ACTIVITIES
Other than the classroom and laboratory learning, following are the suggested student-related *co-curricular* activities which can be undertaken to accelerate the attainment of the various outcomes in this course: Students should conduct following activities in group and prepare reports of about 5 pages for each activity, also collect/record physical evidences for their (student's) portfolio which will be useful for their placement interviews:

a. Prepare journals based on practical performed in laboratory.
b. Library/E-Book survey regarding 'VB.Net' used in software industries.
c. Undertake a survey of different GUI applications and compare with the following points.
   i. Available Applications.
   ii. Application Profile.

11.   SUGGESTED SPECIAL INSTRUCTIONAL STRATEGIES (if any)

These are sample strategies, which the teacher can use to accelerate the attainment of the various outcomes in this course:

a. Massive open online courses (*MOOCs*) may be used to teach various topics/sub topics.
b. *'L' in item No. 4* does not mean only the traditional lecture method, but different types of teaching methods and media that are to be employed to develop the outcomes.
c. About *15-20% of the topics/sub-topics* which is relatively simpler or descriptive in nature is to be given to the students for *self-directed learning* and assess the development of the COs through classroom presentations (see implementation guideline for details).
d. With respect to item No.10, teachers need to ensure to create opportunities and provisions for *co-curricular activities*.
e. Guide student(s) in undertaking micro-projects.
f. Demonstrate students thoroughly before they start doing the practice.
g. Observe continuously and monitor the performance of students in Lab.

12.   SUGGESTED MICRO-PROJECTS

Only one micro-project is planned to be undertaken by a student that needs to be assigned to him/her in the beginning of the semester. In the first four semesters, the micro-project are group-based. However, in the fifth and sixth semesters, it should be preferably be *individually* undertaken to build up the skill and confidence in every student to become problem solver so that s/he contributes to the projects of the industry. In special situations where groups have to be formed for micro-projects, the number of students in the group should *not exceed three.*

The micro-project could be industry application based, internet-based, workshop-based, laboratory-based or field-based. Each micro-project should encompass two or more COs which are in fact, an integration of PrOs, UOs and ADOs. Each student will have to maintain dated work diary consisting of individual contribution in the project work and give a seminar presentation of it before submission. The total duration of the micro-project should not be less than *16 (sixteen) student engagement hours* during the course. The student ought to submit micro-project by the end of the semester to develop the industry oriented COs.

A suggestive list of micro-projects are given here. Similar micro-projects could be added by the concerned faculty:

a. **Hotel Management applications–** Prepare
   - Hotel room booking system having variety of rooms,
   - Hotel billing system for various services used by the guest.
b. **Store Management Application -** Prepare a menu driven application for inventory management in Store.
c. **Students' Record System -**Prepare a menu driven application to Maintain Academic record of students from multiple streams.
d. **Traffic signals control Design-** Design an application for traffic signal control using Timer Control.

13.   SUGGESTED LEARNING RESOURCES

S. No.	Title of Book	Author	Publication
1	Visual Basic .NET The Complete Reference	Jeffrey R. Shapiro	McGraw–Hill, California, USA ISBN0–07–213381–3
2	Visual Basic .NET Programming Black Book	Holzner Steven	Dreamtech Press, 2015, New Delhi, ISBN-13:978-81-7722-609-6.
3	Beginning Visual Basic 2012	Bryan Newsome	Wrox Press, USA, Edition: 2012; ISBN: 9781118311813,
4	GUI Application Development using VB.Net	Dr. Rajendra Kawale	Devraj Publication,

14. SUGGESTED SOFTWARE/LEARNING WEBSITES

a. http://www.vbtutor.net/index.php/visual-basic-2012-tutorial
b. http://howtostartprogramming.com/vb-net
c. https://www.tutorialspoint.com/vb.net
d. http://vb.net-informations.com
e. http://www.java2s.com/Tutorial/VB/CatalogVB.htm
f. http://www.functionx.com/vbnet
g. http://www.dfit.dfinalsolution.com/dotnet%20tutorial%20for%20beginners.pdf

Program Name	: Computer Engineering Program Group
Program Code	: CO/CM/IF/CW
Semester	: Fourth
Course Title	: Java Programming
Course Code	: 22412

1.   RATIONALE

Java is platform independent, open-source object oriented programming language enriched with free and open source libraries. In current industrial scenario Java has the broad industry support and is prerequisite with many allied technologies like Advanced Java, Java Server Pages, and Android Application Development. Thus, current industrial trends necessitate acquiring Java knowledge for Computer Engineering and Information Technology graduates. This course develops necessary skills in students to apply object oriented programming techniques in Java so that students will be able to develop complete applications using core Java.

2.   COMPETENCY

The aim of this course is to help the student to attain the following industry identified competency through various teaching learning experiences:

*   **Develop Applications using Java**.

3.   COURSE OUTCOMES (COs)

The theory, practical experiences and relevant soft skills associated with this course are to be taught and implemented, so that the student demonstrates the following *industry oriented* COs associated with the above mentioned competency:

   a.   Develop programs using Object Oriented methodology in Java.
   b.   Apply concept of inheritance for code reusability.
   c.   Develop programs using multithreading.
   d.   Implement Exception Handling.
   e.   Develop programs using graphics and applet.
   f.   Develop programs for handling I/O and file streams.

4.   TEACHING AND EXAMINATION SCHEME

Teaching Scheme			Credit (L+T+P)	Examination Scheme												
					Theory						Practical					
L	T	P		Paper Hrs.	ESE		PA		Total		ESE		PA		Total	
					Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min
3	-	4	7	3	70	28	30*	00	100	40	50#	20	50	20	100	40

(*): Under the theory PA; Out of 30 marks, 10 marks of theory PA are for micro-project assessment to facilitate integration of COs and the remaining 20 marks is the average of 2 tests to be taken during the semester for the assessment of the UOs required for the attainment of the COs.
Legends: *L-Lecture; T – Tutorial/Teacher Guided Theory Practice; P -Practical; C – Credit, ESE -End Semester Examination; PA - Progressive Assessment.*

5.   COURSE MAP(with sample COs, PrOs, UOs, ADOs and topics)

This course map illustrates an overview of the flow and linkages of the topics at various levels of outcomes (details in subsequent sections) to be attained by the student by the end of the course, in all domains of learning in terms of the industry/employer identified competency depicted at the centre of this map.
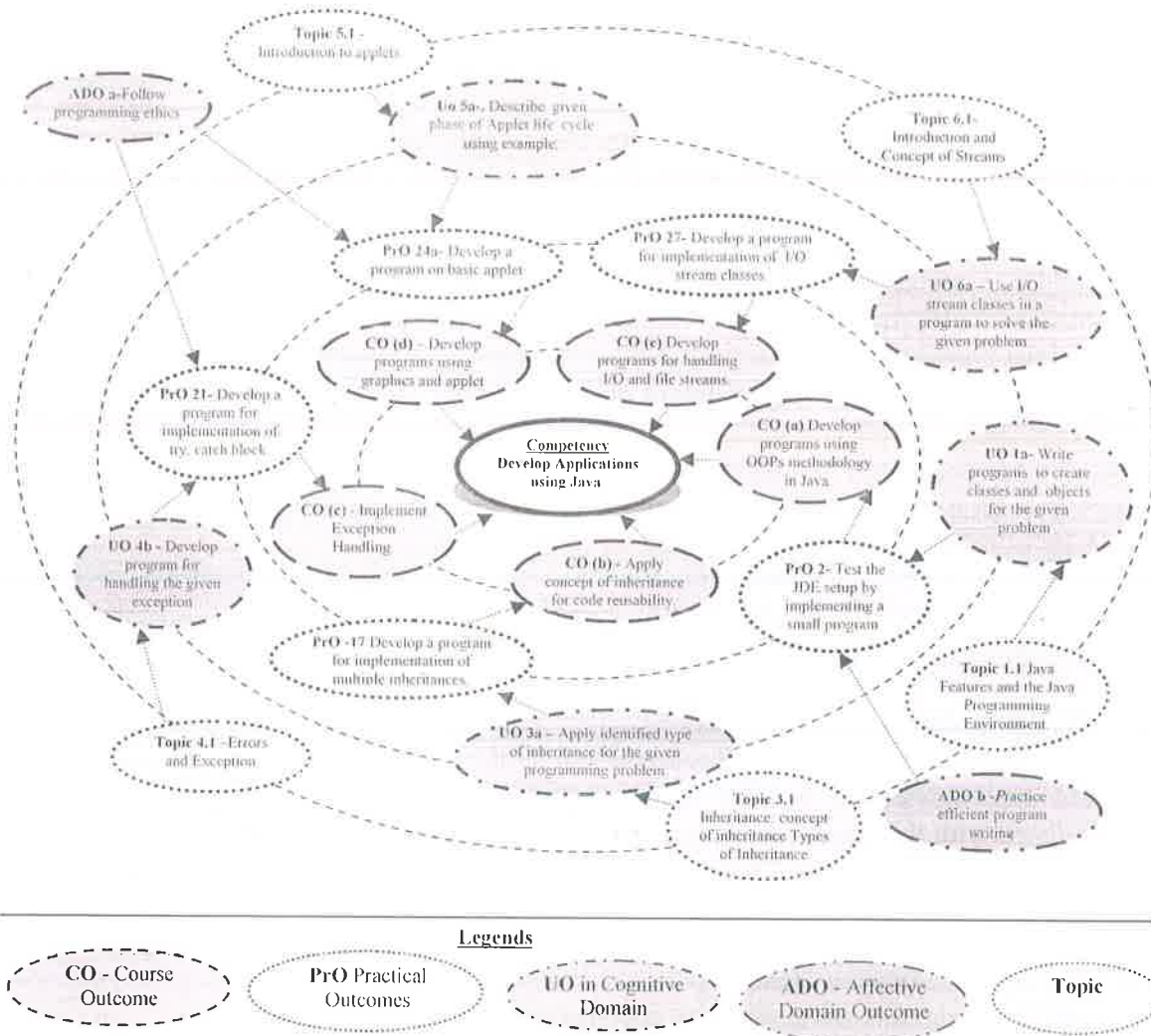

Figure 1 - Course Map

6.   SUGGESTED PRACTICALS/ EXERCISES

The practicals in this section are PrOs (i.e. sub-components of the COs) to be developed and assessed in the student for the attainment of the competency.

S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. Required
1.	Setup a Java Programming development environment by using: a) Command prompt.(Classpath and path setup) b) Any IDE (Eclipse, Jcreator etc.).	I	02*
2.	Test the JDE setup by implementing a small program.	I	02
3.	Develop programs to demonstrate use of if statements and its different forms.	I	02*
4.	Develop programs to demonstrate use of- a) Switch – Case statement b) Conditional if (? :)	I	02*
5.	Develop programs to demonstrate use of Looping Statement 'for'	I	02*
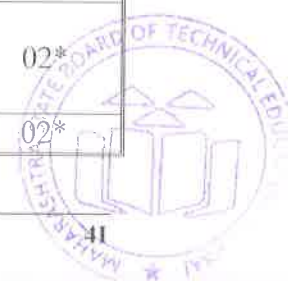
S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. Required
6.	Develop programs to demonstrate use of 'while', 'do-while'	I	02*
7.	Develop a program for implementation of implicit type casting in Java. Part-I	I	02
8.	Develop a program for implementation of implicit type casting in Java. Part-II	I	02
9.	Develop a program for implementation of explicit type conversion in Java.	I	02*
10.	a) Develop a program for implementation of Constructor. b) Develop a program for implementation of multiple constructors in a class.	II	02*
11.	Develop a program for implementation of different functions of String Class. Part-I	II	02
12.	Develop a program for implementation of different functions of String Class. Part-II	II	02
13.	Develop a program for implementation of Arrays in Java.	II	02*
14.	Develop a program for implementation of Vectors in Java.	II	02*
15.	Develop a program for implementation of Wrapper Class to convert primitive into object.	II	02*
16.	Develop a program for implementation of Wrapper Class to convert object into primitive.	II	02*
17.	Develop program which implements the concept of overriding.	II	02*
18.	Develop a program for implementation of Single and Multilevel inheritance.	III	02*
19.	Develop a program for implementation of multiple inheritances.	III	02*
20.	Develop a program to import different classes in package.	III	02*
21.	Develop a program for implementation of multithreading operation Part-I	IV	02*
22.	Develop a program for implementation of multithreading operation Part-II	IV	02
23.	Develop a program for implementation of try, catch block. Part-I	IV	02
24.	Develop a program for implementation of try, catch block. Part-II	IV	02
25.	Develop a program for implementation of try, catch and finally block.	IV	02*
26.	Develop programs for implementation of throw, throws clause. Part-I	IV	02*
27.	Develop programs for implementation of throw, throws clause. Part-II	IV	02*
28.	Develop minimum two basic Applets. Display output with applet viewer and browser. a) Develop a program on basic applet.	V	02*
	b) Develop a program using control loops in applets.	V	02
29.	Write a program to create animated shape using graphics and applets. You may use following shapes: a) Lines and Rectangles. b) Circles and Ellipses.	V	02*
	c) Arcs d) Polygons with fillPolygon method.		02
30.	Develop a program to draw following shapes, graphics and applets. a) Cone b) Cylinders	V	02*

S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. Required
	c) Cube		
	d) Square inside a circle e) Circle inside a square		02
31.	Develop a program for implementation of I/O stream classes.	VI	02*
32.	Develop a program for implementation of file stream classes.	VI	02*
	Total		**64**

Note

i. *A suggestive list of PrOs is given in the above table. More such PrOs can be added to attain the COs and competency. A judicial mix of minimum 24 or more practical need to be performed, out of which, the practicals marked as '*' are compulsory, so that the student reaches the 'Application Level' of Bloom's Taxonomy' as generally required by the industry.*

ii. *The 'Process' and 'Product' related skills associated with each PrO are to be assessed according to a suggested sample given below:*

Sr. No.	Performance Indicators	Weightage in %
1	Representation of classes and objects.	10
2	Coding.	40
4	Testing and Debugging of the Program.	30
5	Correctness of ProgramOutput.	10
6	Submission of report in time.	10
	Total	**100**

The above PrOs also comprise of the following social skills/attitudes which are Affective Domain Outcomes (ADOs) that are best developed through the laboratory/field based experiences:

    a. Follow safety practices.
    b. Practice good housekeeping.
    c. Demonstrate working as a leader/a team member.
    d. Follow ethical practices.

The ADOs are not specific to any one PrO, but are embedded in many PrOs. Hence, the acquisition of the ADOs takes place gradually in the student when s/he undertakes a series of practical experiences over a period of time. Moreover, the level of achievement of the ADOs according to Krathwohl's 'Affective Domain Taxonomy' should gradually increase as planned below:

    • 'Valuing Level' in 1st year
    • 'Organising Level' in 2nd year and
    • 'Characterising Level' in 3rd year.

7. MAJOR EQUIPMENT/ INSTRUMENTS REQUIRED

The major equipment with broad specification mentioned here will usher in uniformity in conduct of experiments, as well as aid to procure equipment by authorities concerned.
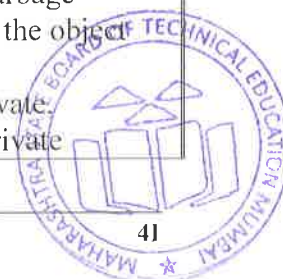
S. No.	Equipment Name with Broad Specifications	Pro S.No.
1	Computer with JDK1.8 or above	All
2	Any IDE for Java Programming such as Eclipse, Jcreator or any other product.	

8.   UNDERPINNING THEORY COMPONENTS

The following topics are to be taught and assessed in order to develop the sample UOs given below for achieving the COs to attain the identified competency. More UOs could be added.

Unit	Unit Outcomes (UOs) (in cognitive domain)	Topics and Sub-topics
Unit – I **Basic Syntactical constructs in Java**	1a. Write programs to create classes and objects for the given problem. 1b. Explain the characteristics of the given Java token. 1c. Explain the function of the given operator with example. 1d. Construct the expressions using implicit and explicit type conversions to solve the given problem. 1e. Develop the programs using relevant control structure to solve the given problem.	1.1 Java Features and the Java Programming Environment, Object Oriented, Compiled, Interpreted, Platform independent, Portable, Robust and Secure, Dynamic. 1.2 Defining a class, creating object, accessing class members 1.3 Java Tokens and Data types, Constants and Symbolic Constants, variables, dynamic initialization, data types, array and string, scope of variable, typecasting, and standard default values. 1.4 Operators and Expressions, Arithmetic Operators, Relational Operators, Logical Operators, Increment and Decrement, Conditional Operators, Bit wise Operators, Instance of Operators, Dot Operators, Operator precedence and associativity, Evaluation of Expressions, Type conversions in expressions, Mathematical Functions - min(),max(), sqrt(), pow(), exp(), round(), abs(). 1.5 Decision making and looping: If statement, if else statement, nested if else statement, if else if ladder, the switch statement, nested switch statement, The ?:operator, The while statement, the Do while statement, the 'for' statement, break, continue and return statement, nested loops, labeled loops, for-each version of the for loop.
Unit-II **Derived Syntactical Constructs in Java**	2a. Use Constructors for the given programming problem. 2b. Identify scope and lifetime of a variable in the given program code. 2c. Describe the given visibility control with example. 2d. Write the programs by implementing arrays to solve	2.1 Constructors and methods, types of constructors, nesting of methods, argument passing the 'this' keyword, command line arguments, varargs: variable-length arguments, garbage collection, finalize() method, the object class. 2.2 Visibility Control Public, Private, Protected, default, friendly private

Unit	Unit Outcomes (UOs) (in cognitive domain)	Topics and Sub-topics
	the given problem. 2e. Develop programs using vectors and wrapper classes for the given problem.	Protected access. 2.3 Arrays and Strings: Types of arrays, creating an array, strings, string classes and string buffer, vectors, wrapper classes, enumerated types.
Unit– III **Inheritance, Interface and Package**	3a. Apply the identified type of inheritance for the given programming problem. 3b. Differentiate between overloading and overriding for the given example. 3c. Develop program using the specified interface. 3d. Create user defined package for the given problem. 3e. Add class and interface to the given package.	3.1 Inheritance: concept of inheritance Types of Inheritance 3.2 Single Inheritance, multilevel Inheritance, Hierarchical Inheritance, method and constructor overloading and overriding, dynamic method dispatch, final variables, final methods, use of super, abstract methods and classes, static members. 3.3 Interfaces: Define Interface, implementing interface ,accessing interface, variables and methods, extending interfaces, interface references, nested interfaces 3.4 Package: Define package, type of package naming and creating packages, accessing package, import statement, static import, adding class and interfaces to a package.
Unit– IV **Exception Handling and Multithreading**	4a. Distinguish the errors and exceptions (if any) in the given example. 4b. Develop program for handling the given exception. 4c. Crate threads to run the given multiple processes in the given program. 4d. Explain the function of the specified phase in thread life cycle using the given example.	4.1 Errors and Exception :Types of errors, exceptions, try and catch statement, nested try statement, throws and Finally statement, build-in exceptions, chained exceptions, creating own exception(throw clause), subclasses. 4.2 Multithreaded Programming Creating a Thread: By extending to thread class and by implementing runnable Interface, Life cycle of thread: Thread Methods:wait(), sleep(), notify(), resume(), suspend(), stop().Thread exceptions, thread priority and methods, synchronization, inter-thread communication, deadlock.
Unit –V **Java Applets and Graphics Programming**	5a. Describe the given phase of applet life cycle using a typical example. 5b. Develop programs using applet implementation for the given problem. 5c. Develop program for implementing the given geometric shape. 5d. Develop program for implementing the given font	5.1 Introduction to applets: Applet, Applet life cycle (skeleton), Applet tag, Adding Applet to HTML file, passing parameter to applet, embedding <applet> tags in java code, adding controls to applets. 5.2 Graphics Programming: Graphics classes, lines, rectangles, ellipse, circle, arcs, polygons, color and fonts, setColor(), getColor(), setForeGround(), setBackGround(), font class, variable defined by font class: name, pointSize,

Unit	Unit Outcomes (UOs) (in cognitive domain)	Topics and Sub-topics
	settings.	size, style, font methods: getFamily(), getFont(), getFontname (), getSize(), getStyle(), getAllFonts() and get available font family name() of the graphics environment class.
Unit –VI Managing Input /Output/ Files in Java	6a. Use I/O stream classes in a program to solve the given problem. 6b. Write programs for reading and writing character streams to and from the given files. 6c. Write programs for reading and writing bytes to and from the given files. 6d. Write program to demonstrate use of primitive Data types with the specified stream.	6.1 Introduction and Concept of Streams. 6.2 Stream Classes. 6.3 Byte Stream Classes: Input Stream Classes, Output Stream Classes. 6.4 Character Stream Classes, Using streams. 6.5 Using File Class: I/O Exceptions, Creation of Files, Reading/Writing characters, Reading/Writing Bytes, Handling Primitive Data types.

Note: To attain the COs and competency, above listed UOs need to be undertaken to achieve the 'Application Level' of Bloom's 'Cognitive Domain Taxonomy'.

9.   SUGGESTED SPECIFICATION TABLE FORQUESTION PAPER DESIGN

Unit No.	Unit Title	Teaching Hours	Distribution of Theory Marks			
			R Level	U Level	A Level	Total Marks
I	Basic Syntactical constructs in Java	06	02	04	04	10
II	Derived Syntactical Constructs in Java	10	02	06	10	18
III	Inheritance, Interface and Package	10	02	04	06	12
IV	Exception Handling and Multithreading	08	02	04	06	12
V	Java Applets and Graphics Programming	08	02	04	04	10
VI	Managing Input/Output/Files in Java	06	02	02	04	08
	Total	**48**	**12**	**24**	**34**	**70**

Legends: R=Remember, U=Understand, A=Apply and above (Bloom's Revised taxonomy)
Note: This specification table provides general guidelines to assist students for their learning and to teachers to teach and assess students with respect to attainment of UOs. The actual distribution of marks at different taxonomy levels (of R, U and A) in the question paper may vary from above table.

10.   SUGGESTED STUDENT ACTIVITIES
Other than the classroom and laboratory learning, following are the suggested student-related *co-curricular* activities which can be undertaken to accelerate the attainment of the various outcomes in this course: Students should conduct following activities in group and prepare reports of about 5 pages for each activity, also collect/record physical evidences for their (student's) portfolio which will be useful for their placement interviews:

a. Prepare journals based on practical performed in laboratory.
b. Follow coding standards.
c. Develop variety of programs to improve the logical skills.
d. Develop Application oriented real world programs.
e. Prepare power point presentation or animation for understanding different Object Oriented Concepts.

11.   SUGGESTED SPECIAL INSTRUCTIONAL STRATEGIES (if any)

These are sample strategies, which the teacher can use to accelerate the attainment of the various outcomes in this course:

a. Massive open online courses (*MOOCs*) may be used to teach various topics/sub topics.
b. *'L' in item No. 4* does not mean only the traditional lecture method, but different types of teaching methods and media that are to be employed to develop the outcomes.
c. About *15-20% of the topics/sub-topics* which is relatively simpler or descriptive in nature is to be given to the students for *self-directed learning* and assess the development of the COs through classroom presentations (see implementation guideline for details).
d. With respect to item No.10, teachers need to ensure to create opportunities and provisions for *co-curricular activities*.
e. Use different Audio Visual media for Concept understanding.
f. Guide student(s) in undertaking micro-projects.
g. Demonstrate students thoroughly before they start doing the practice.
h. Observe continuously and monitor the performance of students in Lab.

12.   SUGGESTED MICRO-PROJECTS

Only one micro-project is planned to be undertaken by a student that needs to be assigned to him/her in the beginning of the semester. In the first four semesters, the micro-project are group-based. However, in the fifth and sixth semesters, it should be preferably be *individually* undertaken to build up the skill and confidence in every student to become problem solver so that s/he contributes to the projects of the industry. In special situations where groups have to be formed for micro-projects, the number of students in the group should *not exceed three.*

The micro-project could be industry application based, internet-based, workshop-based, laboratory-based or field-based. Each micro-project should encompass two or more COs which are in fact, an integration of PrOs, UOs and ADOs. Each student will have to maintain dated work diary consisting of individual contribution in the project work and give a seminar presentation of it before submission. The total duration of the micro-project should not be less than *16 (sixteen) student engagement hours* during the course. The student ought to submit micro-project by the end of the semester to develop the industry oriented COs.

A suggestive list of micro-projects are given here. Similar micro-projects could be added by the concerned faculty:

a. Mini Banking System for handling deposits and withdrawal.
b. Medical Store stock Management System.
c. Library book issue Management System.
d. Bus Reservation System.
e. Attendance Management System.
f. Develop a small animation using applet, graphics and multithreading.

GUIDELINES FOR DEVELOPING MICRO PROJECTS:

i.   Declare four to five classes and may include Interfaces if required.
ii.   Must use Most of the Object Oriented Concepts.

iii.   Must implement concepts of Inheritance and Exception Handling.
iv.   Must Create Own Package.
v.   May use the constructor overloading and overriding.
vi.   May Use Multithreading if required.

13.   SUGGESTED LEARNING RESOURCES

S. No.	Title of Book	Author	Publication
1.	Programming with JAVA	Balagurusamy E.	Mcgraw Hill Education (India) Private Limited, New Delhi, 5th Edition ISBN-13: 978-93-5134-320-2
2.	Java 8 Programming Black Book	DT Editorial Services	Dreamtech Press, New Delhi, ISBN:978-93-5119-758-4
3.	Java Complete Reference	Schildt Herbert	Mcgraw Hill Education, New Delhi ISBN:9789339212094
4.	Advanced Java Programming	Roy Uttam K	Oxford University Press, New Delhi ISBN :0-19-945550-3
5.	Jawa Programming	Dr. Rajendra Kawale	Devraj Publication

14.   SUGGESTED SOFTWARE/LEARNING WEBSITES

   a.   https://docs.oracle.com/javase/8/docs/
   b.   http://www.nptelvideos.com/java/java_video_lectures_tutorials.php
   c.   http://nptel.ac.in/courses/106105084/25
   d.   http://www.iitk.ac.in/esc101/08Jul/notes.html

Program Name	: **Computer Engineering Program Group**
Program Code	**: CO/CM/IF/CW**
Semester	**: Fourth**
Course Title	**: Software Engineering**
Course Code	**: 22413**

1.   RATIONALE

Software Engineering is the foundation for professional processes to be followed involving principles, techniques, and practices for software development. The course provides a framework for software professionals for building quality assured software products. It enables students to blend the domain specific knowledge with the programming skills to create quality software products.

2.   COMPETENCY

The aim of this course is to help the student to attain the following industry identified competency through various teaching learning experiences:

- **Use relevant software process model for developing software products.**

3.   COURSE OUTCOMES (COs)

The theory, practical experiences and relevant soft skills associated with this course are to be taught and implemented, so that the student demonstrates the following *industry oriented* COs associated with the above-mentioned competency:

- a.   Select suitable Software Process model for software development.
- b.   Prepare software requirement specifications.
- c.   Use Software modeling to create data designs.
- d.   Estimate size and cost of software product.
- e.   Apply project management and quality assurance principles in software development.

4.   TEACHING AND EXAMINATION SCHEME

Teaching Scheme			Credit (L+T+P)	Examination Scheme											
				Paper Hrs.	Theory						Practical					
L	T	P			ESE		PA		Total		ESE		PA		Total	
					Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min
3	-	2	5	3	70	28	30*	00	100	40	25@	10	25	10	50	20

(*): Under the theory PA; Out of 30 marks, 10 marks of theory PA are for micro-project assessment to facilitate integration of COs and the remaining 20 marks is the average of 2 tests to be taken during the semester for the assessment of the UOs required for the attainment of the COs.
Legends: L-*Lecture;* ***T** – Tutorial/Teacher Guided Theory Practice;* ***P** - Practical;* ***C** – Credit,* ***ESE** - End Semester Examination;* ***PA** - Progressive Assessment*

5.   COURSE MAP (with sample COs, PrOs, UOs, ADOs and topics)

This course map illustrates an overview of the flow and linkages of the topics at various levels of outcomes (details in subsequent sections) to be attained by the student by the end of the course. in all domains of learning in terms of the industry/employer identified competency depicted at the centre of this map.
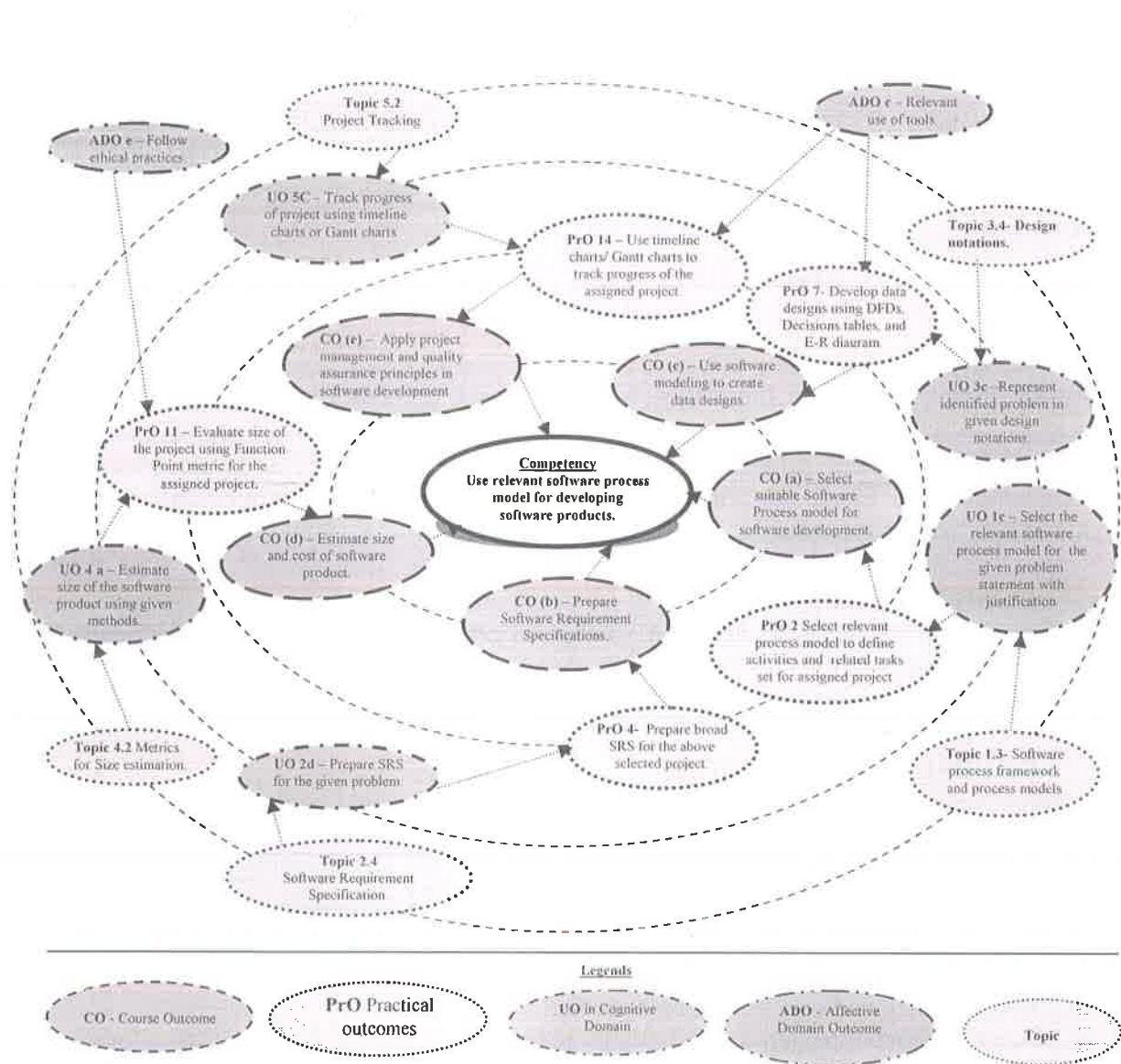
Figure 1 - Course Map

6.   SUGGESTED PRACTICALS/ EXERCISES

The practicals in this section are PrOs (i.e. sub-components of the COs) to be developed and assessed in the student for the attainment of the competency.

S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. Required
1	Write problem statement to define the project title with bounded scope of the project.	I	02*
2	Select relevant process model to define activities and related tasks set for assigned project.	I	02*
3	Gather application specific requirements for assimilate into RE (Requirements engineering) model.	II	02*
4	Prepare broad SRS (software requirement software) for the above selected project.	II	02*
5	Prepare use-cases and draw use – case diagram using Software Modeling Tool.	II	02*

S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. Required
6	Develop the activity diagram to represent flow from one activity to another for software development.	II	02
7	Develop data designs using DFDs (data flow diagram), Decision tables and E-R (entity-relationship) diagram.	III	02*
8	Draw class diagram, Sequence diagram, Collaboration diagram, State Transition Diagram for the assigned project.	III	02
9	Write test cases to validate requirements of assigned project from SRS document.	III	02*
10	Identify risks involved in the project and prepare RMMM (RMMM-Risk Management, Mitigation and Monitoring) plan.	IV	02
11	Evaluate size of the project using Function point metric for the assigned project.	IV	02*
12	Estimate cost of the project using COCOMO (Constructive Cost Model) / COCOMO II approach for the assigned project.	IV	02*
13	Use CPM (Critical Path Method) / PERT (Programme Evaluation and Review Technique) for scheduling the assigned project.	V	02*
14	Use Timeline charts/ Gantt charts to track progress of the assigned project.	V	02
15	Prepare SQA plan that facilitates various attributes of quality of process.	V	02*
16	Prepare SQA plan that facilitates various attributes of quality of product.	V	02*
	Total		**32**

Note

i. *To carry out above listed practical /tasks, relevant* software tool may be chosen (preferably open-source based).

ii. *A suggestive list of PrOs is given in the above table. More such PrOs can be added to attain the COs and competency. A judicial mix of minimum 12 or more practical need to be performed, out of which, the practicals marked as '*' are compulsory, so that the student reaches the 'Application Level' of Bloom's Taxonomy' as generally required by the industry.*

iii. *The 'Process' and 'Product' related skills associated with each PrO is to be assessed according to a suggested sample given below:*

i.

S.No.	Performance Indicators	Weightage in %
1	Problem selection and its feasibility study	20
2	Logical thinking to decompose problem into modules	30
3	Ability to Estimate size and cost of a software	30
4	Presentation and technical documentation skills	10
5	Submission of reports within time.	10
	Total	**100**

The above PrOs also comprise of the following social skills/attitudes which are Affective Domain Outcomes (ADOs) that are best developed through the laboratory/field based experiences:

a. Follow safety practices.
b. Practice good housekeeping.
c. Demonstrate working as a leader/a team member.

    d. Follow ethical practices.

The ADOs are not specific to any one PrO, but are embedded in many PrOs. Hence, the acquisition of the ADOs takes place gradually in the student when s/he undertakes a series of practical experiences over a period of time. Moreover, the level of achievement of the ADOs according to Krathwohl's 'Affective Domain Taxonomy' should gradually increase as planned below:

- 'Valuing Level' in 1st year
- 'Organising Level' in 2nd year and
- 'Characterising Level' in 3rd year.

7.   MAJOR EQUIPMENT/ INSTRUMENTS REQUIRED

The major equipment with broad specification mentioned here will usher in uniformity in conduct of experiments, as well as aid to procure equipment by authorities concerned.

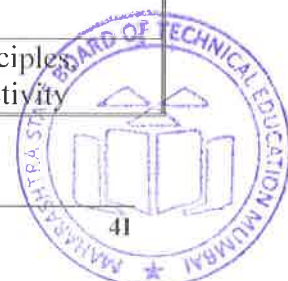| S. No.	Equipment Name with Broad Specifications	Pro. S. No.
1	Hardware: Personal computer. (i3-i5 preferable), RAM minimum 2 GB	For all Experiments
2	Operating system: Windows 7/Windows 8/Windows 10/LINUX or any other .	
3	Software tools: Any UML tool	

8.   UNDERPINNING THEORY COMPONENTS

The following topics are to be taught and assessed in order to develop the sample UOs given below for achieving the COs to attain the identified competency. More UOs could be added.

Unit	Unit Outcomes (UOs) (in cognitive domain)	Topics and Sub-topics
Unit – I Software Development Process	1a. Suggest the attributes that match with standards for the given software application. 1b. Recommend the relevant software solution for the given problem with justification. 1c. Select the relevant software process model for the given problem statement with justification. 1d. Suggest the relevant activities in Agile Development Process in the given situation with justification	1.1 Software, Software Engineering as layered approach and its characteristics, Types of software. · 1.2 Software development framework. 1.3 Software Process Framework, Process models: Perspective Process Models, Specialized Process Models. 1.4 Agile Software development: Agile Process and its importance, Extreme Programming, Adaptive Software Development, Scrum, Dynamic Systems Development Method (DSDM), Crystal. 1.5 Selection criteria for software process model.
Unit– II Software Requirement Engineering	2a. Apply the principles of software engineering for the given problem. 2b. Choose the relevant	2.1 Software Engineering Practices and its importance, Core principles. 2.2 Communication Practices, Planning Practices, Modelling practices.

Unit	Unit Outcomes (UOs) (in cognitive domain)	Topics and Sub-topics
	'requirement engineering' steps in the given problem. 2c. Represent the 'requirement engineering' model in the given problem. 2d. Prepare SRS for the given problem.	construction practices, software deployment (Statement and meaning of each principle for each practice). 2.3 Requirement Engineering: Requirement Gathering and Analysis, Types of requirements (Functional, Product, organizational, External Requirements), Eliciting Requirements, Developing Use-cases, Building requirement models, Requirement Negotiation, Validation. 2.4 Software Requirement Specification: Need of SRS, Format, and its Characteristics.
Unit– III Software Modelling and Design	3a. Identify the elements of analysis model for the given software requirements. 3b. Apply the specified design feature for software requirements modeling. 3c. Represent the specified problem in the given design notation. 3d. Explain the given characteristics of software testing. 3e. Prepare test cases for the given module.	3.1 Translating Requirement model into design model: Data Modelling. 3.2 Analysis Modelling: Elements of Analysis model. 3.3 Design modelling: Fundamental Design Concepts (Abstraction, Information hiding, Structure, Modularity, Concurrency, Verification, Aesthetics). 3.4 Design notations: Data Flow Diagram (DFD), Structured Flowcharts, Decision Tables. 3.5 Testing – Meaning and purpose, testing methods - Black-box and White-box, Level of testing – Unit testing. 3.6 Test Documentation – Test Case Template, test plan, Introduction to defect report, test summary report.
Unit-IV Software Project Estimation	4a. Estimate the size of the software product using the given method. 4b. Estimate the cost of the software product using the given empirical method. 4c. Evaluate the size of the given software using CoCoMo model. 4d. Apply the RMMM strategy in Identified risks for the given software development problem.	4.1 The management spectrum – 4P's 4.2 Metrics for Size Estimation: Line of Code(LoC), Function Points(FP). 4.3 Project Cost Estimation Approaches: Overview of Heuristic, Analytical, and Empirical Estimation. 4.4 COCOMO (Constructive Cost Model), COCOMO II. 4.5 Risk Management: Risk Identification, Risk Assessment, Risk Containment, RMMM strategy.
Unit –V Software	5a. Use the given scheduling technique for the	5.1 Project Scheduling: Basic principles Work breakdown structure, Activity

Unit	Unit Outcomes (UOs) (in cognitive domain)	Topics and Sub-topics
quality assurance and Security	identified project. 5b. Draw the activity network for the given task. 5c. Prepare the timeline chart/ Gantt chart to track progress of the given project. 5d. Describe the given Software Quality Assurance (SQA) activity. 5e. Describe features of the given software quality evaluation standard.	network and critical path Method, Scheduling techniques (CPM, PERT). 5.2 Project Tracking: Timeline charts, Earned Value Analysis, Gantt Charts 5.3 Software Quality Management vs. Software Quality Assurance. Phases of Software Quality Assurance: Planning, Activities, audit, and review 5.4 Quality Evaluation standards: Six Sigma, ISO for software, CMMI: Levels, Process areas. 5.5 Software Security,Introduction to DevOps,Secure software engineering

Note: To attain the COs and competency, above listed UOs need to be undertaken to achieve the 'Application Level' of Bloom's 'Cognitive Domain Taxonomy'.

9. SUGGESTED SPECIFICATION TABLE FOR QUESTION PAPER DESIGN

Unit No.	Unit Title	Teaching Hours	Distribution of Theory Marks			
			R Level	U Level	A Level	Total Marks
I	Software development process	08	04	04	04	12
II	Software Requirement Engineering	10	02	04	08	14
III	Software Modelling and Design	10	-	04	10	14
IV	Software Project Estimation	10	04	04	08	16
V	Software Project Management and quality assurance	10	04	04	06	14
	Total	**48**	**14**	**20**	**36**	**70**

Legends: R=Remember, U=Understand, A=Apply and above (Bloom's Revised taxonomy)
__Note__: This specification table provides general guidelines to assist student for their learning and to teachers to teach and assess students with respect to attainment of UOs. The actual distribution of marks at different taxonomy levels (of R, U and A) in the question paper may vary from above table.
This specification table also provides a general guideline for teachers to frame internal end semester practical theory exam paper which students have to undertake.

10. SUGGESTED STUDENT ACTIVITIES

Other than the classroom and laboratory learning, following are the suggested student-related *co-curricular* activities which can be undertaken to accelerate the attainment of the various outcomes in this course: Students should conduct following activities in group and prepare reports of about 5 pages for each activity, also collect/record physical evidences for their (student's) portfolio which will be useful for their placement interviews:

a. Prepare journals based on practical performed in laboratory.
b. Give seminar on relevant topics.
c. Study and analyze college website from perspective of software application.
d. Study and analyze any available application software from perspective of software engineering.

11.   SUGGESTED SPECIAL INSTRUCTIONAL STRATEGIES (if any)

These are sample strategies, which the teacher can use to accelerate the attainment of the various outcomes in this course:

a. Massive open online courses (*MOOCs*) may be used to teach various topics/sub topics.

b. *'L' in item No. 4* does not mean only the traditional lecture method, but different types of teaching methods and media that are to be employed to develop the outcomes.

c. About *15-20% of the topics/sub-topics* which is relatively simpler or descriptive in nature is to be given to the students for *self-directed learning* and assess the development of the COs through classroom presentations (see implementation guideline for details).

d. With respect to item No.10, teachers need to ensure to create opportunities and provisions for *co-curricular activities*.

e. Guide student(s) in undertaking micro-projects.

f. Demonstrate students thoroughly before they start doing the practice.

g. Encourage students to refer different websites to have deeper understanding of the subject.

h. Observe continuously and monitor the performance of students.

12.   SUGGESTED MICRO-PROJECTS

Only one micro-project is planned to be undertaken by a student that needs to be assigned to him/her in the beginning of the semester. In the first four semesters, the micro-project are group-based. However, in the fifth and sixth semesters, it should be preferably be *individually* undertaken to build up the skill and confidence in every student to become problem solver so that s/he contributes to the projects of the industry. In special situations where groups have to be formed for micro-projects, the number of students in the group should *not exceed three.*

The micro-project could be industry application based, internet-based, workshop-based, laboratory-based or field-based. Each micro-project should encompass two or more COs which are in fact, an integration of PrOs, UOs and ADOs. Each student will have to maintain dated work diary consisting of individual contribution in the project work and give a seminar presentation of it before submission. The total duration of the micro-project should not be less than *16 (sixteen) student engagement hours* during the course. The student ought to submit micro-project by the end of the semester to develop the industry oriented COs.

A suggestive list of micro-projects are given here. Similar micro-projects could be added by the concerned faculty:

a. Study and analyze given software and write the characteristics and functions of the same.

b. Case study of application specific software product for requirement engineering
   i.   Identify the problem statement
   ii. Perform feasibility analysis
   iii. Identify application specific requirements by following RE steps
   iv. Prepare SRS

c. Choose any problem statement and use data models to represent the solution
   i.   Search and utilize different UML tools to represent models

d. Choose a problem, create activity network and use different project scheduling and tracking tools for the same.

13.   SUGGESTED LEARNING RESOURCES

S. No.	Title of Book	Author	Publication
1	Software Engineering: A practitioner's approach	Pressman, Roger S.	McGraw Hill Higher Education, New Delhi, (Seventh Edition)ISBN 978–0–07–337597–7
2	Software Engineering Concepts	Fairly, Richard	McGraw Hill Education New Delhi – 2001, ISBN-13: 9780074631218
3	Software Engineering: Principles and practices	Jain, Deepak	Oxford University Press, New Delhi ISBN 9780195694840

14.   SUGGESTED SOFTWARE/LEARNING WEBSITES

- a. http://www.rspa.com/spi/
- b. www.tutorialspoint.com//software_engineering/
- c. www.versionone.com/agile-101/
- d. www.sei.cmu.edu
- e. www.nptel.ac.in/courses/
- f. https://techbeacon.com/secure-devops

Program Name	: Computer Engineering Program Group
Program Code	: CO/CM/IF/CW
Semester	: Fourth
Course Title	: Data Communication and Computer Network
Course Code	: 22414

1.   RATIONALE

A data communication and computer networks has been growing with rapid technological progress. Computer communication through networking becomes essential part of our life. By considering importance of networking in day today life, it is essential for students to know the basic concept of networks like network classification, network topologies, network devices. This course deal with the important concepts and techniques related to data communication and enable students to have an insight in to technology involved to make the network communication possible.

2.   COMPETENCY

The aim of this course is to help the student to attain the following *industry identified* competency through various teaching learning experiences:

- **Maintain data communication and computer network**

3.   COURSE OUTCOMES (COs)

The theory, practical experiences and relevant soft skills associated with this course are to be taught and implemented, so that the student demonstrates the following industry oriented COs associated with the above mentioned competency:

a.   Analyze the functioning of data communication and computer network.
b.   Select relevant transmission media and switching techniques as per need.
c.   Analyse the transmission errors with respect to IEEE standards.
d.   Configure various networking devices.
e.   Configure different TCP/IP services.

4.   TEACHING AND EXAMINATION SCHEME

Teaching Scheme			Credit (L+T+P)	Examination Scheme												
				Paper Hrs.	Theory						Practical					
L	T	P			ESE		PA		Total		ESE		PA		Total	
					Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min
4	-	2	6	3	70	28	30*	00	100	40	25@	10	25	10	50	20

(*): Under the theory PA; Out of 30 marks, 10 marks of theory PA are for micro-project assessment to facilitate integration of COs and the remaining 20 marks is the average of 2 tests to be taken during the semester for the assessment of the UOs required for the attainment of the COs.
Legends: L*-Lecture;* ***T*** *– Tutorial/Teacher Guided Theory Practice;* ***P*** *- Practical;* ***C*** *– Credit,* ***ESE*** *- End Semester Examination;* ***PA*** *- Progressive Assessment.*

5.   COURSE MAP (with sample COs, PrOs, UOs, ADOs and topics)

This course map illustrates an overview of the flow and linkages of the topics at various levels of outcomes (details in subsequent sections) to be attained by the student by the end of the

course, in all domains of learning in terms of the industry/employer identified competency depicted at the centre of this map.

Figure 1 - Course Map

6.   SUGGESTED PRACTICALS/ EXERCISES

The practicals in this section are PrOs (i.e. sub-components of the COs) to be developed and assessed in the student for the attainment of the competency.

S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. Required
1.	Configure Peer-to-Peer Network with at least three hosts.	I	02*
2.	Create desired standard network cable including cross cable and test by using cableTester	II	02*
3.	Connect Computers using given topology with wired media	III	02*
4.	Connect Computers using wireless media	III	02
5.	Write a C program for CRC Error Detection.	III	02
6.	Create a Network Using Bluetooth-(Piconet/Scatternet)	III	02
7.	Share Printer and Folder in a network and transfer a file from one computer to another.	IV	02

S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. Required
8.	Install operating system(Windows/Linux-RedHat/Ubuntu)	IV	02*
9.	Configure File Server	IV	02
10.	Configure Client To File Server and use file services.	IV	02
11.	Configure Static and Dynamic IP addresses	V	02*
12.	Configure DHCP server.	V	02*
13.	Run Basic TCP/IP Utilities and Network commands : ipconfig, ping , tracert, netstat, pathping, route	V	02*
14.	Install Wireshark and configure as packet sniffer	V	02
15.	Set access rights and security permissions for user	V	02
16.	Create IPV6 based small computer network using a simulator (prferebaly open source based simulator)	V	02
17.	Setting up a wireless network	IV	02
	Total		**34**

Note

i. *A suggestive list of PrOs is given in the above table. More such PrOs can be added to attain the COs and competency. A judicial mix of minimum 24 or more practical need to be performed, out of which, the practicals marked as '*' are compulsory, so that the student reaches the 'Precision Level' of Dave's 'Psychomotor Domain Taxonomy' as generally required by the industry.*

ii. *The 'Process' and 'Product' related skills associated with each PrO is to be assessed according to a suggested sample given below:*

S. No.	Performance Indicators	Weightage in %
a.	Effective practical implementation within specified time	60
b.	Effective handling of network component	10
c.	Answer to sample questions	20
d.	Submit report in time	10
	Total	**100**

The above PrOs also comprise of the following social skills/attitudes which are Affective Domain Outcomes (ADOs) that are best developed through the laboratory/field based experiences:

    a. Follow safety practices.
    b. Handle network components carefully.
    c. Demonstrate working as a leader/a team member.
    d. Follow ethical practices.

The ADOs are not specific to any one PrO, but are embedded in many PrOs. Hence, the acquisition of the ADOs takes place gradually in the student when s/he undertakes a series of practical experiences over a period of time. Moreover, the level of achievement of the ADOs according to Krathwohl's 'Affective Domain Taxonomy' should gradually increase as planned below:

- 'Valuing Level' in 1st year
- 'Organising Level' in 2nd year and
- 'Characterising Level' in 3rd year.

7.　MAJOR EQUIPMENT/ INSTRUMENTS REQUIRED

The major equipment with broad specification mentioned here will usher in uniformity in conduct of experiments, as well as aid to procure equipment by authorities concerned.

S. No.	Equipment Name with Broad Specifications	PrO. S. No.
1.1	Computer system (Any computer system with basic configuration)	All
1.2	Network connecting device, transmission media	All
1.3	Network cable Tester, crimping tool, RJ-45 connectors, Ethernet cable	2
1.4	Wireshark sniffing tool	15

8.   UNDERPINNING THEORY COMPONENTS

The following topics are to be taught and assessed in order to develop the sample UOs given below for achieving the COs to attain the identified competency. More UOs could be added:

Unit	Unit Outcomes (UOs) (in cognitive domain)	Topics and Sub-topics
Unit – I Fundamentals of Data Communication and Computer Network	1a. Describe role of the given component in the process of data communication. 1b. Compare the characteristic of analog and digital signals on the given parameter. 1c. Explain the process of data communication using the given mode. 1d. Classify computer networks on the specified parameter. 1e. Select network architecture for the given situation with justificsation.	1.1 Process of data communication and its components: Transmitter, Receiver, Medium, Message, Protocol. 1.2 Protocols, Standards, Standard organizations.Bandwidth, Data Transmission Rate, Baud Rate and Bits per second. 1.3 Modes of Communication (Simplex, Half duplex, Full Duplex). 1.4 Analog Signal and Digital Signal, Analog and Digital transmission:Analog To Digital ,Digital To Analog Conversion 1.5 Fundamental Of Computer Network: Definition And Need Of Computer Network,Applications,Network Benefits. 1.6 Classification Of Network: LAN,WAN,MAN 1.7 Network Architechture:Peer To Peer, Client Server Network
Unit-II Transmission Media and Switching	2a. Explain with sketches the construction of given type of cable. 2b. Explain with sketches the characteristics of the given type of unguided transmission media. 2c. Explain with sketches the working of the given Multiplexing technique. 2d. Describe with sketches the working principle of the given switching technique.	2.1 Communication Media: Guided Transmission Media Twisted-Pair Cable, Coaxial Cable Fiber-Optic Cable 2.2 Unguided Transmssion Media Radio Waves, Microwaves, Infrared, Satellite 2.3 Line-of-Sight Transmission Point to Point, Broadca 2.4 Multiplexing: Frequency-Division Multiplexing Time -Division Multiplexing 2.5 Switching: Circuit-switched networks

Unit	Unit Outcomes (UOs) (in cognitive domain)	Topics and Sub-topics
	2e. Compare different Switching techniques on the given parameter.	Packet -swtiched networks
Unit– III **Error** **Detection,C** **orrection** **and** **Wireless** **Communica** **tion**	3a. Explain working of the given error detection and correction method. 3b. Explain features of the given IEEE communication standard. 3c. Explain characteristics of the given layer in IEEE 802.11 architecture. 3d. Compare the specified generations of mobile telephone system on the given parameter. 3e. Explain with sketches the process of creating Bluetooth environment using the given architechture.	3.1 Types of Errors:Single Bit Error and Burst Error, Redundancy 3.2 Error Detection:Longitudinal Redundancy Check(LRC),Vertical Redundancy Check(VRC),Cyclic Redundancy Check(CRC)Forward 3.3 Error Correction: Forward error Correction 3.3 IEEE standards: 802.1, 802.2, 802.3, 802.4, 802.5 3.4 Wireless LANs: 802.11 Architecture, MAC Sublayer,Addressing Mechanism 3.5 Bluetooth Architecture: Piconet, Scatternet 3.6 Mobile Generations: 1G, 2G, 3G, 4G and 5G
Unit– IV **Network** **Topologies** **And** **Network** **Devices**	4a. Identify relavent network topology for the given situation. 4b. Compare different topologies on the given parameter. 4c. Select network connecting device for the given situation. 4d. Describe with sketches the procedure to configure the given networking device.	4.1 Network Topologies : Introduction, Definition, Selection, Criteria, Types of Topology- i) Bus ii) Ring iii) Star iv)Mesh v)Tree vi)Hybrid 4.2 Network Connecting Devices:Hub, Switch, Router, Repeater, Bridge, Gateway, Modem, Wireless infrastructure Components
Unit –V **Reference** **Models**	5a. Identify functions and features of the given layer of OSI Reference model. 5b. Compare the specified service on the given parameters. 5c. Classify IP Addresses on the basis of its class from the given set of addresses. 5d. Distinguish between IPv4 and IPv6 on the given parameters. 5e. Describe with sketches the	5.1 OSI Reference Model: Layered Architecture , Peer-to- Peer Processes-Interfaces between Layer, Protocols, Organization of the Layers, Encapsulation Layers of the OSI Reference Model (Functions and features of each Layer) – Physical Layer, Data-Link Layer, Network Layer, Transport Layer, Session Layer, Presentation Layer, Application Layer 5.2 TCP/IP Model: Layered Architechture Data Link Layer:Nodes and links,services,two categoriesof links,two

Unit	Unit Outcomes (UOs) (in cognitive domain)	Topics and Sub-topics
	procedure to configure the given TCP/IP service.	sub layers, Link layer addressing:three types of addresses,address resolution protocal(ARP), Network Layer: Addresses: address space,classful and classless addeessing, dyanamic host configuration protocol(DHCP), network address resolution(NAT). Transport layer protocal:transport layer services, connectionless and conection oriented protocol. 5.3 Introduction –Addressing mechanism in the Internet IP Addressing – IP Address classes, classless IP addressing, Subnetting, supernetting, Masking, 5.4 IPv4 and IPv6 5.5 OSI and TCP / IP Network Model.

Note: To attain the COs and competency, above listed UOs need to be undertaken to achieve the 'Application Level' of Bloom's 'Cognitive Domain Taxonomy'.

9.   SUGGESTED SPECIFICATION TABLE FOR QUESTION PAPER DESIGN

Unit No.	Unit Title	Teaching Hours	Distribution of Theory Marks			
			R Level	U Level	A Level	Total Marks
I	Fundamental Of Data Communication And Computer Network	10	04	04	04	12
II	Transmission Media and Switching	14	04	06	06	16
III	Error Detection,Correction and Wireless Communication	14	02	04	06	12
IV	Network Topologies And Network Devices	10	02	04	04	10
V	Reference Models	16	06	06	08	20
	Total	**64**	**18**	**24**	**28**	**70**

Legends: R=Remember, U=Understand, A=Apply and above (Bloom's Revised taxonomy)
Note: This specification table provides general guidelines to assist students for their learning and to teachers to teach and assess students with respect to attainment of LOs. The actual distribution of marks at different taxonomy levels (of R, U and A) in the question paper may vary from above table.

10.   SUGGESTED STUDENT ACTIVITIES

Other than the classroom and laboratory learning, following are the suggested student-related *co-curricular* activities which can be undertaken to accelerate the attainment of the various outcomes in this course: Students should conduct following activities in group and prepare reports of about 5 pages for each activity, also collect/record physical evidences for their (student's) portfolio which will be useful for their placement interviews:

    A. Prepare specification table for Guided media and Unguided media.
    B. Classify network connecting devices with their specifications.

11.   SUGGESTED SPECIAL INSTRUCTIONAL STRATEGIES (if any)

These are sample strategies, which the teacher can use to accelerate the attainment of the various outcomes in this course:

a. Massive open online courses (*MOOCs*) may be used to teach various topics/sub topics.

b. *'L' in item No. 4* does not mean only the traditional lecture method, but different types of teaching methods and media that are to be employed to develop the outcomes.

c. About *15-20% of the topics/sub-topics* which is relatively simpler or descriptive in nature is to be given to the students for *self-directed learning* and assess the development of the COs through classroom presentations (see implementation guideline for details).

d. With respect to item No.10, teachers need to ensure to create opportunities and provisions for *co-curricular activities*.

e. Use animations to explain various network topologies,OSI Layers,network connecting devices.

f. Guide student(s) in undertaking micro-projects

12.   SUGGESTED MICRO-PROJECTS

Only one micro-project is planned to be undertaken by a student that needs to be assigned to him/her in the beginning of the semester. In the first four semesters, the micro-project are group-based. However, in the fifth and sixth semesters, it should be preferably be individually undertaken to build up the skill and confidence in every student to become problem solver so that s/he contributes to the projects of the industry. In special situations where groups have to be formed for micro-projects, the number of students in the group should not exceed three.

The micro-project could be industry application based, internet-based, workshop-based, laboratory-based or field-based. Each micro-project should encompass two or more COs which are in fact, an integration of PrOs, UOs and ADOs. Each student will have to maintain dated work diary consisting of individual contribution in the project work and give a seminar presentation of it before submission. The total duration of the micro-project should not be less than 16 (sixteen) student engagement hours during the course. The student ought to submit micro-project by the end of the semester to develop the industry oriented COs.

A suggestive list of micro-projects are given here. Similar micro-projects could be added by the concerned faculty:

a. Create a small Network install, configure various devices and perform at least one peer-to-peer service and client/server service over it.

b. Prepare a report on recent and widely used Unguided media in industries depending on Cost, speed, efficiency, reliability.

c. Design layout of a Network for department, Deciding upon type of network, number/length of components with their specifications.

13.   SUGGESTED LEARNING RESOURCES

S. No.	Title of Book	Author	Publication
1	Data communications and networking.	Forouzan Behrouz A.	Tata McGraw Hill, New Delhi, 2006, ISBN : 9780-07-296775-3
2	Computer Networks	Tanenbaum Andrew S.	PHI Learning Pvt Ltd, Delhi ISBN-13: 978-0-13-212695-3

S. No.	Title of Book	Author	Publication
3	Data Communication and Networks	Godbole Achyut	Tata McGraw Hill, New Delhi, 2006, ISBN : 0070472971
4	Internetworking with TCP/IP Principles, Protocols and Architechtures	Comer Douglas E.	PHI Learning Pvt Ltd, Delhi ISBN: 81-203-2065-4
5	Computer Networking	T. M. Bansod	---

14. SUGGESTED SOFTWARE/LEARNING WEBSITES

a. www.nptelvideos.in/2012/11/data-communication.html
b. http://www.myreadingroom.co.in/notes-and-studymaterial/68-dcn/750-analog-to-analog-conversion-techniques.html
c. http://www.tutorial-reports.com/wireless/wlanwifi/wifi_architecture.php
d. http://standards.ieee.org/about/get/802/802.11.html
e. www.tutorialspoint.com/data_communication_computer_network/
f. http://www.studytonight.com/computer-networks/overview-of-computer-networks
g. http://whirlpool.net.au/wiki/windows_nw_diag_cmds
h. http://nptel.ac.in/downloads/106105080/
i. http://scanftree.com/programs/c/c-program-to-implement-crc-cyclic-redundancy-code/

Program Name	: **Computer Engineering Program Group**
Program Code	: **CO/CM/CW**
Semester	: **Fourth**
Course Title	: **Microprocessors**
Course Code	: **22415**

1.   RATIONALE

Microprocessor is the main component of computer where 8086 is the base of all upward developed processors till current processors. This course will cover the basics of 8086 and its architecture along with instruction set, assembly language programming with effective use of procedure and macros. This course also covers the architectural issues such as instruction set program and data types. On top that, the students are also introduced to the increasingly important area of parallel organization. This subject serves as a basic to develop hardware related projects. This course will enable the students to inculcate assembly language programming concepts and methodology to solve problems.

2.   COMPETENCY

The aim of this course is to help the student to attain the following *industry identified* competency through various teaching learning experiences:

- **Develop assembly level language programming using 8086.**

3.   COURSE OUTCOMES (COs)

The theory, practical experiences and relevant soft skills associated with this course are to be taught and implemented, so that the student demonstrates the following industry oriented COs associated with the above mentioned competency:

- a. Analyze the functional block of 8086 microprocessor.
- b. Write assembly language program for the given problem.
- c. Use instructions for different addressing modes.
- d. Develop an assembly language program using assembler.
- e. Develop assembly language programs using procedures, macros and modular programming approach.

4.   TEACHING AND EXAMINATION SCHEME

Teaching Scheme			Credit (L+T+P)	Examination Scheme											
				Paper Hrs.	Theory						Practical					
					ESE		PA		Total		ESE		PA		Total	
L	T	P			Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min
4	-	2	6	3	70	28	30*	00	100	40	25@	10	25	10	50	20

(*): Under the theory PA; Out of 30 marks, 10 marks of theory PA are for micro-project assessment to facilitate integration of COs and the remaining 20 marks is the average of 2 tests to be taken during the semester for the assessment of the UOs required for the attainment of the COs.

Legends: L-Lecture; **T** – Tutorial/Teacher Guided Theory Practice; **P** - Practical; **C** – Credit; **ESE** - End Semester Examination; **PA** - Progressive Assessment, **'#'**: No Theory Examination

5.   COURSE MAP(with sample COs, PrOs, UOs, ADOs and topics)

This course map illustrates an overview of the flow and linkages of the topics at various levels of outcomes (details in subsequent sections) to be attained by the student by the end of the course, in all domains of learning in terms of the industry/employer identified competency depicted at the centre of this map.
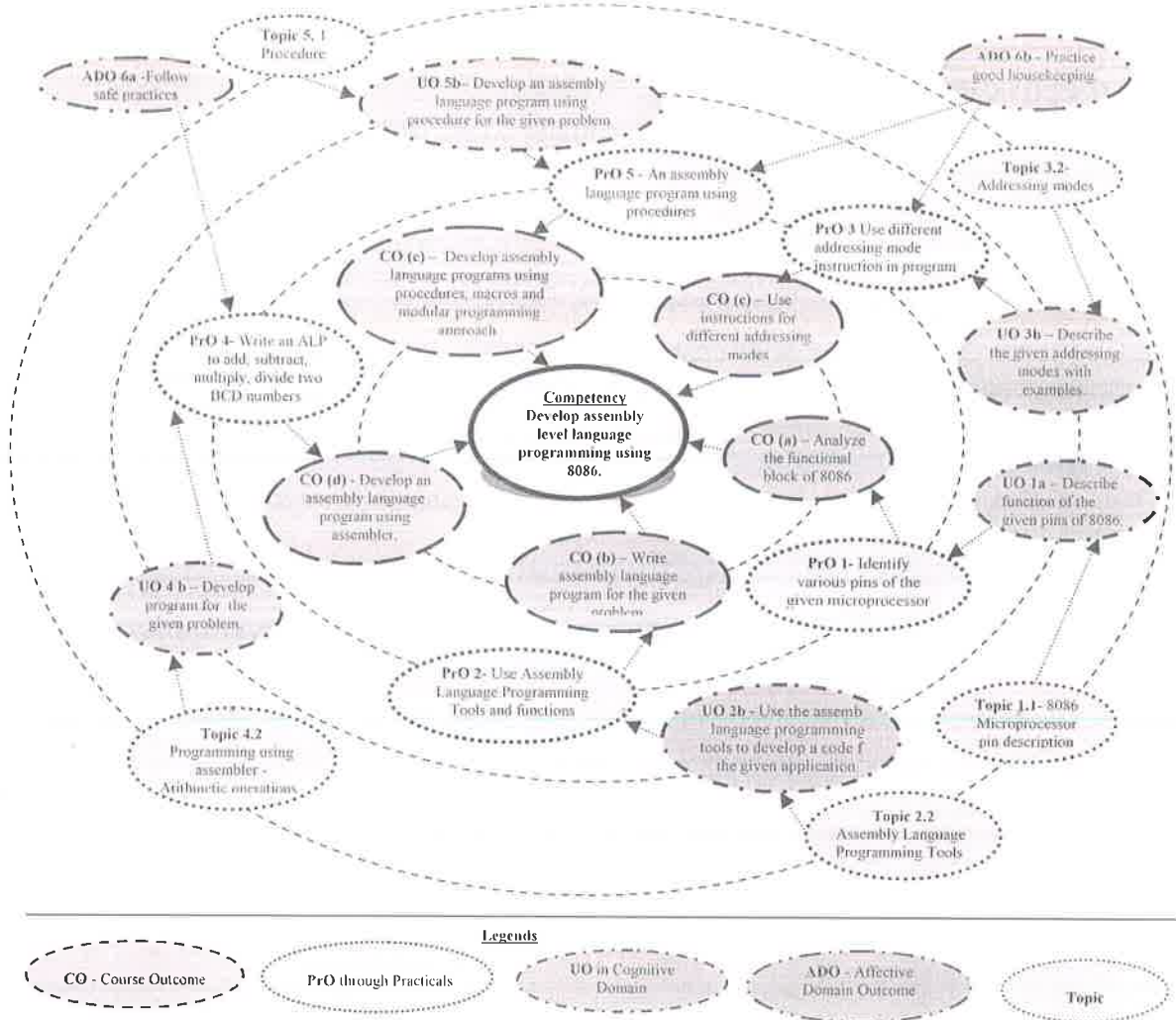

Figure 1 - Course Map

6.   SUGGESTED PRACTICALS/ EXERCISES

The practicals in this section are PrOs (i.e. sub-components of the COs) to be developed and assessed in the student for the attainment of the competency.

S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. Required
1	Identify various pins of the given microprocessor.	I	02*
2	Use Assembly Language Programming Tools and functions	II	02*
3	Use different addressing mode instruction in program (a) Write an Assembly Language Program (ALP) to add two given 8 and 16 bit numbers.	III	02*

S. No.	Practical Outcomes (PrOs)	Unit No.	Approx. Hrs. Required
	(b) Write an Assembly Language Program (ALP) to subtract two given 8 and 16 bit numbers.		
4	(a) Write an ALP to multiply two given 8 and 16 bit unsigned numbers. (b) Write an ALP to multiply two given 8 and 16 bit signed numbers.	III	02
5	(a) Write an ALP to perform block transfer data using string instructions (b) Write an ALP to perform block transfer data without using string instructions.	III	02
6	(a) Write an ALP to compare two strings without using string instructions. (b) Write an ALP to compare two strings using string instructions	III	02
7	(a) Write an ALP to divide two unsigned numbers (b) Write an ALP to divide two signed numbers	III	02
8	Write an ALP to add, subtract, multiply, divide two BCD numbers.	IV	02
9	Implement loop in assembly language program (a) Write an ALP to find sum of series of Hexadecimal Numbers. (b) Write an ALP to find sum of series of BCD numbers.	IV	02*
10	(a) Write an ALP to find smallest number from array of n numbers. (b) Write an ALP to find largest number from array of n numbers.	IV	02 *
11	(a) Write an ALP to arrange numbers in array in ascending order. (b) Write an ALP to arrange numbers in array in descending order.	IV	02
12	(a) Write an ALP to arrange string in reverse order (b) Write an ALP to find string length. (c) Write an ALP to concatenation of two strings.	IV	02
13	(a) Write an ALP to check a given number is ODD or EVEN. (b) Write an ALP to count ODD and/or EVEN numbers in array.	IV	02
14	(a) Write an ALP to check a given number is POSITIVE or NEGATIVE (b)Write an ALP to count POSITIVE and/or NEGATIVE numbers in array.	IV	02
15	(a) Write an ALP to count number of '1' in a given number (b) Write an ALP to count number of '0' in a given number	IV	02
16	An assembly language program using procedures (a) Write an ALP for addition, subtraction, multiplication and division. (b) Write an ALP using procedure to solve equation such as Z = (A+B)*(C+D)	V	02*
17	Write an assembly language program using macros. (a) Write an ALP for addition, subtraction, multiplication and division. (b) Write an ALP using MACRO to solve equation such as 01Z =	V	02*

S. No.	Practical Outcomes (PrOs)		Unit No.	Approx. Hrs. Required
	(A+B)*(C+D)			
		Total		34

Note

i. *A suggestive list of PrOs is given in the above table. More such PrOs can be added to attain the COs and competency. A judicial mix of minimum 24 or more practical need to be performed, out of which, the practicals marked as '*' are compulsory, so that the student reaches the 'Precision Level' of Dave's 'Psychomotor Domain Taxonomy' as generally required by the industry.*

ii. *The 'Process' and 'Product' related skills associated with each PrO is to be assessed according to a suggested sample given below:*

S.No.	Performance Indicators	Weightage in %
1	Able to write algorithm and draw flow chart.	20
2	Use Assembly language programming tools to create, edit, assemble and link the assembly language programs.	40
3	Debug, test and execute the programs	20
4	Able to answer oral questions.	10
5	Submission of report in time.	10
	Total	100

The above PrOs also comprise of the following social skills/attitudes which are Affective Domain Outcomes (ADOs) that are best developed through the laboratory/field based experiences:

    a. Follow safety practices.
    b. Practice good housekeeping.
    c. Demonstrate working as a leader/a team member.
    d. Follow ethical practices.

The ADOs are not specific to any one PrO, but are embedded in many PrOs. Hence, the acquisition of the ADOs takes place gradually in the student when s/he undertakes a series of practical experiences over a period of time. Moreover, the level of achievement of the ADOs according to Krathwohl's 'Affective Domain Taxonomy' should gradually increase as planned below:

    • 'Valuing Level' in 1st year
    • 'Organizing Level' in 2nd year and
    • 'Characterizing Level' in 3rd year.

7.   MAJOR EQUIPMENT/ INSTRUMENTS REQUIRED

The major equipment with broad specification mentioned here will usher in uniformity in conduct of experiments, as well as aid to procure equipment by authorities concerned.

S. No.	Equipment Name with Broad Specifications	PrO. S. No.
1	Hardware: Personal computer, (i3-i5 preferable), RAM minimum 2 GiB onwards.	For all Experiments

S. No.	Equipment Name with Broad Specifications	PrO. S. No.
2	Operating system: Windows XP/Windows 7 onward	
3	Software: Editor: EDIT, NOTEPAD Assembler: TASM/MASM Linker: TLINK/LINK Debugger: TD/Debug of Windows Operating System	

8.   UNDERPINNING THEORY COMPONENTS

The following topics are to be taught and assessed in order to develop the sample UOs given below for achieving the COs to attain the identified competency. More UOs could be added.

Unit	Unit Outcomes (UOs) (in cognitive domain)	Topics and Sub-topics
Unit – I **8086-** **16 Bit** **Microproc essor**	1a. Describe function of the given pin of 8086. 1b. Explain with sketches the working of given unit in 8086 microprocessor. 1c. State functions of the given registers of 8086 microprocessor. 1d. Calculate the physical address for the given segmentation of 8086 microprocessor.	1.1 8086 Microprocessor: Salient features, Pin descriptions 1.2 Architecture of 8086: Functional Block diagram, Register organization 1.3 Concepts of pipelining 1.4 Memory segmentation, Physical memory addresses generation
Unit– II **The Art of** **Assembly** **Language** **Programm ing**	2a. Describe the given steps of program development /execution. 2b. Write steps to develop a code for the given problem using assembly language programming. 2c. Use relevant command of debugger to correct the specified programming error. 2d. Describe function of the given assembler directives with example.	2.1 Program development steps:Defining problem and constrains, Writing Algorithms, Flowchart, Initialization checklist, Choosing instructions, Converting algorithms to assembly language programs 2.2 Assembly Language Programming Tools: Editors, Assembler, Linker, Debugger 2.3 Assembler directives
Unit– III **Instruction** **Set of 8086** **Microproc essor**	3a. Determine the length of the given instruction. 3b. Describe the given addressing modes with examples. 3c. Explain the operation performed by the given instruction during its execution. 3d. Identify the addressing modes in the given instructions.	3.1 Machine Language Instruction format 3.2 Addressing modes 3.3 Instruction set, Groups of Instructions: Arithmetic instructions, Logical Instructions, Data transfer instructions, Bit manipulation instructions, String Operation instructions, Program control transfer or branching instructions. Process control instructions

Unit	Unit Outcomes (UOs) (in cognitive domain)	Topics and Sub-topics
Unit-IV Assembly Language Programming	4a. Use the given model of assembly language programs for the given problem. 4b. Develop the relevant program for the given problem. 4c. Apply relevant control loops in the program for the given problem. 4d. Use string instructions for the given strings/block to manipulate its elements.	4.1 Model of 8086 assembly language programs 4.2 Programming using assembler : Arithmetic operations on Hex and BCD numbers, Sum of Series, Smallest and Largest numbers from array, Sorting numbers in Ascending and Descending order, Finding ODD,EVEN, Positive and Negative numbers in the array, Block transfer, String Operations - Length, Reverse, Compare, Concatenation, Copy, Count Numbers of '1' and '0' in 16 bit number.
Unit –V Procedure and Macro	5a. Apply the relevant 'parameter-passing' method in the given situation. 5b. Develop an assembly language program using the relevant procedure for the given problem. 5c. Develop an assembly language program using MACROS for the given problem. 5d. Compare procedures and macros on the basis of the given parameters.	5.1 Procedure: Defining and calling Procedure - PROC, ENDP, FAR and NEAR Directives; CALL and RET instructions; Parameter passing methods, Assembly Language Programs using Procedure 5.2 Macro: Defining Macros, MACRO and ENDM Directives, Macro with parameters, Assembly Language Programs using Macros

Note: *To attain the COs and competency, above listed UOs need to be undertaken to achieve the 'Application Level' of Bloom's 'Cognitive Domain Taxonomy'.*

9. SUGGESTED SPECIFICATION TABLE FOR QUESTION PAPER DESIGN

Unit No.	Unit Title	Teaching Hours	Distribution of Theory Marks			
			R Level	U Level	A Level	Total Marks
I	8086- 16 Bit Microprocessor	08	02	02	10	14
II	The Art of Assembly Language Programming	12	-	02	06	08
III	Instruction Set of 8086 Microprocessor	16	02	04	10	16
IV	Assembly Language Programming	16	02	02	16	20
V	Procedure and Macro	12	02	02	08	12
Total		**64**	**8**	**12**	**50**	**70**

Legends: R=Remember, U=Understand, A=Apply and above (Bloom's Revised taxonomy)
Note: This specification table provides general guidelines to assist students for their learning and to teachers to teach and assess students with respect to attainment of UOs. The actual

distribution of marks at different taxonomy levels (of R, U and A) in the question paper may vary from above table.

10.   SUGGESTED STUDENT ACTIVITIES

Other than the classroom and laboratory learning, following are the suggested student-related **co-curricular** activities which can be undertaken to accelerate the attainment of the various outcomes in this course: Students should conduct following activities in group and prepare reports of about 5 pages for each activity, also collect/record physical evidences for their (student's) portfolio which will be useful for their placement interviews:

- a. Prepare journals based on practical performed in laboratory.
- b. Library/E-Book survey regarding assembly language programming used in Computer industries.
- c. Prepare power point presentation for showing different types of Assembly language Programming Applications.

11.   SUGGESTED SPECIAL INSTRUCTIONAL STRATEGIES (if any)

These are sample strategies, which the teacher can use to accelerate the attainment of the various outcomes in this course:

- a. Massive open online courses (**MOOCs**) may be used to teach various topics/sub topics.
- b. **'L' in item No. 4** does not mean only the traditional lecture method, but different types of teaching methods and media that are to be employed to develop the outcomes.
- c. About **15-20% of the topics/sub-topics** which is relatively simpler or descriptive in nature is to be given to the students for **self-directed learning** and assess the development of the COs through classroom presentations (see implementation guideline for details).
- d. With respect to item No.10, teachers need to ensure to create opportunities and provisions for **co-curricular activities**.
- e. Guide student(s) in undertaking micro-projects.
- f. No. of practical's selection to be performed should cover all units.

12.   SUGGESTED MICRO-PROJECTS

Only one micro-project is planned to be undertaken by a student that needs to be assigned to him/her in the beginning of the semester. In the first four semesters, the micro-project are group-based. However, in the fifth and sixth semesters, it should be preferably be individually undertaken to build up the skill and confidence in every student to become problem solver so that s/he contributes to the projects of the industry. In special situations where groups have to be formed for micro-projects, the number of students in the group should not exceed three.

      The micro-project could be industry application based, internet-based, workshop-based, laboratory-based or field-based. Each micro-project should encompass two or more COs which are in fact, an integration of PrOs, UOs and ADOs. Each student will have to maintain dated work diary consisting of individual contribution in the project work and give a seminar presentation of it before submission. The total duration of the micro-project should not be less than 16 (sixteen) student engagement hours during the course. The student ought to submit micro-project by the end of the semester to develop the industry oriented COs.

      A suggestive list of micro-projects are given here. Similar micro-projects could be added by the concerned faculty:

- a. **Conversion of Number System** - Each group will develop a program to perform following operations (Any One):
  - i. Convert Hexadecimal number to equivalent BCD.

        ii. Convert BCD number to Equivalent Hexadecimal Number

b. **Array** - Each group will develop a program to perform following operations (Any One):

        i. Separate ODD and EVEN number from given array, store them in separate array and find the sum.

        ii. Separate ODD and EVEN number from given array, store them in separate array and find the smallest or largest among them.

        iii. Separate ODD and EVEN number from given array, store them in separate array and sort numbers in ascending or descending order.

c. **Basic mathematic functions** - Each group will develop a program to perform following operations (Any One):

        i. Generate Fibonacci Series

        ii. Find Factorial of Number

d. **String Manipulation project** - Each group will develop a program to perform following operations (Any One):

        i. Convert lower case string to upper case string and vice versa.

        ii. Check string for Palindromes.

        iii. Search given character in string; find how many times it is present in string and its position.

13.   SUGGESTED LEARNING RESOURCES

S. No.	Title of Book	Author	Publication
1	Microprocessor and interfacing (programming and hardware)	Hall, Douglas V.	McGraw Hill Education, New Delhi, 2015, ISBN-13: 978-0070257429
2	The 8088 and 8086 Microprocessors	Triebel, Walter A., Singh, Avtar	Pearson Publications, New Delhi, 2015, ISBN 13: 9780130930811
3	Microprocessors and Microcontrollers	Latha, C., Murugeshwari, B.	SCITECH Publications, Chennai, 2015, ISBN: 978-81-8371-702-1

14.   SUGGESTED SOFTWARE/LEARNING WEBSITES

a. Assembler TASM/MASM, Linker LINK/TLINK, Debugger OS Debuge/TD

b. www.intel.com

c. www.pcguide.com/ref/CPU

d. www.CPU-World.com/Arch/

e. www.techsource .com / engineering- parts/microprocessor.html

f. https://www.elprocus.com/8051-assembly-language-programming/

g. https://www.tutorialspoint.com/assembly_programming/

h. http://www.slideshare.net/search/slideshow?searchfrom=headerandq=assembly+language+programming+of+8086andud=anyandft=allandlang=**andsort=