

DEPARTMENT OF COMPUTER SCIENCE
Teaching Plan for the Academic Year 2023-2024
Semester I

Programme Educational Objectives (PEOs)

PEO	Upon completion of UG Degree Programme, the graduates will be able to:
PEO – 1	apply appropriate theoretical knowledge to participate in activities that support humanity and economic development nationally and globally, developing as leaders in their fields of expertise.
PEO – 2	pursue lifelong learning and continuous improvement of the knowledge and skills with the highest professional and ethical standards.
PEO – 3	inculcate practical knowledge for developing professional empowerment and entrepreneurship and societal services.

Programme Outcomes (POs)

PO	Upon completion of B.Sc. Degree Programme, the graduates will be able to:
PO – 1	utilize scientific knowledge to pursue higher studies in the relevant field.
PO – 2	create innovative ideas to enhance entrepreneurial skills for economic independence.
PO – 3	face challenging competitive examinations that offer rewarding careers.
PO – 4	reflect upon green initiatives and take responsible steps to build a sustainable environment.
PO – 5	handle ethical issues with social responsibility.
PO – 6	communicate effectively and collaborate successfully with peers to become competent professionals.

Programme Specific Outcomes (PSOs)

PSOs	Upon completion of the B.Sc. Degree Programme, the graduates will be able to:
PSO – 1	obtain sufficient knowledge and skills enabling them to undertake further studies in Computer Science and its allied areas on multiple disciplines linked with Computer Science.
PSO - 2	evaluate and apply emerging technologies in computer science to develop innovative solutions for real-world problems
PSO – 3	develop a range of generic skills helpful in team building, problem solving, technical ability, employment, internships, communication and societal activities.
PSO - 4	communicate effectively, work collaboratively, and demonstrate ethical and professional attitudes in diverse settings.
PSO - 5	sensitize various economic issues related to Development, Growth, International Economics, Sustainable Development and Environment

Department : Computer Science
Class : I M. Sc Computer Science
Title of the Course : Core Course I: Analysis and Design of Algorithm
Semester : I
Course Code : SP231CC1

Course Code	L	T	P	Credits	Inst. Hours	Total Hours	Marks		
							CIA	External	Total
SP231CC1	6	-	-	5	6	90	25	75	100

Objectives

1. Enable the students to learn the Elementary Data Structures and algorithms.
2. Presents an introduction to the algorithms, their analysis and design.
3. Discuss various methods like Basic Traversal and Search Techniques, divide and conquer method, Dynamic programming, backtracking.
4. Understood the various design and analysis of the algorithms.

Course Outcomes

CO	Upon completion of this course, the students will be able to:	PSO addressed	Cognitive level
CO – 1	get knowledge about algorithms and determines their time complexity.	PSO – 1	K2(U)
CO – 2	gain good understanding of Greedy method and its algorithm.	PSO – 2	K3(U)
CO – 3	able to describe about graphs using dynamic programming technique.	PSO – 4	K4(An)
CO – 4	demonstrate the concept of backtracking & branch and bound technique.	PSO – 5	K3(Ap)
CO – 5	explore the traversal and searching technique and apply it for trees and graphs.	PSO – 3	K6(C)

Teaching plan

Total Contact hours: 90 (Including lectures, assignments and tests)

Unit	Module	Topic	Teaching Hours	Cognitive level	Pedagogy	Assessment/ Evaluation
I	Introduction					
	1.	Introduction about Algorithm, Algorithm Definition	3	K1(U)	Introductory session	Overview
	2.	Algorithm Specification	2	K2(R)	Problem solving	Simple definitions
	3.	Space complexity, Time Complexity, Asymptotic Notations	2	K2(U)	Lecture using Chalk and talk	Solve problems
	4.	Elementary Data Structures, Stacks and Queues	3	K2(U)	PPT	Finish a procedure in many steps
	5.	Binary Tree, Binary Search Tree	3	K2(U)	Lecture using Chalk and talk	Evaluation essay
	6.	Heap sort	1	K3(Ap)	Demonstration	Concept definitions
	7.	Graph	1	K3(Ap)	Problem solving	Problem solving questions
II	Traversal And Search Techniques					
	1.	Basic Traversal And Search Techniques	3	K1(R)	Lecture using Chalk and talk	Evaluation through short test
	2.	Techniques for Binary Trees	2	K2(U)	Demonstration	Map knowledge
	3.	Techniques for Graphs	3	K2(U)	Lecture using videos	Differentiate between various ideas

	4.	Divide and Conquer: General Method Binary Search	3	K3(Ap)	PPT	Seminar
	5.	Merge Sort	2	K3(Ap)	Demonstration	Recall steps
	6.	Quick Sort	2	K3(Ap)	Demonstration	MCQ
III	Greedy Method					
	1.	The Greedy Method Introduction	3	K2(U)	PPT	Short essays
	2.	General Method	3	K1(R)	Lecture using Chalk and talk	MCQ
	3.	Knapsack Problem	3	K5(E)	Demonstration	Recall steps
	4.	Minimum Cost Spanning Tree	3	K4(An)	Lecture using videos	True/False
	5.	Single Source Shortest Path	3	K3(Ap)	Demonstration	MCQ
IV	Dynamic Programming					
	1.	Dynamic Programming General Method	2	K2(U)	Lecture using Chalk and talk	Short summary
	2.	Multistage Graphs	2	K3(Ap)	Lecture using videos	Concept explanations
	3.	All Pair Shortest Path	2	K3(Ap)	Demonstration	Recall Steps
	4.	Optimal Binary Search Trees	3	K4(An)	PPT	Evaluation through short test
	5.	0/1 Knapsacks	2	K3(Ap)	Lecture using Chalk and talk	Short summary
	6.	Traveling Salesman Problem	2	K6(C)	Problem solving	Evaluation through short test

	7.	Flow Shop Scheduling	2	K5(E)	Demonstration	Concept explanations
V	Backtracking					
	1.	Backtracking General Method	2	K1(R)	Lecture using Chalk and talk	True/False
	2.	8-Queens Problem	2	K3(Ap)	Demonstration	Evaluation through problems
	3.	Sum of Subsets	2	K3(Ap)	Problem solving	Recall Steps
	4.	Graph Coloring	3	K4(An)	Demonstration	MCQ
	5.	Hamiltonian Cycles	2	K3(Ap)	Problem solving	Short essays
	6.	Branch And Bound: The Method	2	K2(U)	Lecture using Chalk and talk	Seminar
	7.	Traveling Salesperson	2	K5(E)	PPT	Evaluation through short test
VI	1.	Expert Lectures, Online Seminars, Webinars	2	K2(U)	PPT	Concept explanations

Course Focussing on Employability/ Entrepreneurship/ Skill Development: Skill Development

Activities (Em/ En/SD): 1. Solve problems for Traveling Salesman.

2. Solve problems for Graph Coloring.

3. Apply Quick sort algorithm for a set of numbers.

Assignment: Heap Sort, Multistage Graphs

Seminar Topic: Binary Search, Branch and Bound

Sample questions

Part A (1 Mark)

1. $O(n)$ is ____

a) Linear

b) Quadrate

c) Cubic

d) Exponential

2. A heap is a ____ tree
 - a) Binary
 - b) Completely Binary
 - c) Almost Completely Binary
 - d) None of the above
3. The complexity of merge sort is ____
 - a) (n)
 - b) (n logn)
 - c) (n²)
 - d) (logn)
4. In the greedy method.
 - a) Only one decision sequence is generated
 - b) Two decision sequence is generated
 - c) Many decision sequences are generated
 - d) None of the above
5. Which of the following is not a backtracking algorithm?
 - a) Knight tour problem
 - b) N Queens problem
 - c) Towers of Hanoi
 - d) M-Coloring

Part B (4 Marks)

6. Write about Stack.
7. Is quick sort a stable sorting method? Justify.
8. What do you understand about Knapsack problem?
9. Write notes on multistage graphs.
10. Compare BFS and DFS algorithm with an example graph and denote its time complexities.

Part C (8 Marks)

11. Explain about Algorithm Specifications.
12. Write about the techniques for Binary trees.
13. Apply greedy algorithm to generate single-source shortest path with an example graph. Mention its time complexity.
14. Explain Flow shop scheduling.
15. Describe the algorithm for Hamiltonian cycles and determine the order of magnitude of the worst case computing time for the backtracking procedure that finds all Hamiltonian cycles.

Head of the Department
Mrs.J.Anto Hepzie Bai

Course Instructor
Dr.S.Immaculate Shyla

Department : Computer Science
Class : I M.Sc Computer Science
Title of the Course : Core Course II: Object Oriented Analysis and Design & C++
Semester : I
Course Code : SP231CC2

Course Code	L	T	P	Credits	Inst. Hours	Total Hours	Marks		
							CIA	External	Total
SP231CC2	6	-	-	4	6	90	25	75	100

Objectives

1. Present the object model, classes and objects, object orientation, machine view and model management view.
2. Enables the students to learn the basic function, principles and concepts of object-oriented analysis and design.
3. Enable the students to understand C++ language with respect to OOAD

Course Outcomes

CO	Upon completion of this course, the students will be able to:	PSO addressed	Cognitive level
CO - 1	understand the concept of Object-Oriented development and modelling techniques	PSO - 1	K1, K2
CO - 2	gain knowledge about the various steps performed during object design	PSO - 3	K2, K3
CO - 3	abstract object-based views for generic software systems	PSO - 2	K3
CO - 4	link OOAD with C++ language	PSO - 4	K4, K5
CO - 5	apply the basic concept of OOPs and familiarize to write C++ program	PSO - 5	K5, K6

Teaching plan

Total Contact hours: 90 (Including lectures, assignments and tests)

Unit	Module	Topic	Teaching Hours	Cognitive level	Pedagogy	Assessment/ Evaluation
I	The Object Model:					
	1.	The Object Model: Introduction, The Evolution of the Object Model	4	K2(U)	Lecture using Chalk and talk, Demonstration, PPT.	Evaluation through short test, MCQ.
	2.	Elements of the Object Model	2	K1(R)	Lecture using Chalk, PPT	Recall steps, Concept definitions
	3.	Applying the Object Model. Classes and Objects	4	K3(Ap)	Lecture using Chalk and talk, Demonstration, PPT	Questioning, Discussions
	4.	The Nature of an Object	2	K4(An)	Lecture using videos, PPT	Seminar
	5.	Relationship among Objects.	3	K5(E)	Lecture, Group discussion	Short summary MCQ
II	Classes and Object					
	1	Classes and Object: Nature of Class	3	K1(R)	Lecture with illustrations	Check knowledge by asking questions.
	2	The Interplay of classes and Objects.	2	K2(U)	Lecture cum Demonstration, PPT	Evaluation through short test
	3	Classification: The importance of Proper Classification	4	K4(An)	Lecture using Chalk and talk, Group Discussion, PPT	Short summary, slip test
	4	Identifying classes and objects	3	K5(E)	Lecture using videos, Demonstration, PPT,	Concept explanations, Short summary
	5	Key Abstractions and Mechanism.	3	K3(A)	Lecture using Chalk and talk, Group Discussion, PPT	Evaluation through online Quiz, Home work
III	Introduction to C++					
	1	Introduction to C++	3	K2(U)	Lecture using Chalk and talk, Group Discussion,	test, MCQ, True/False, , Concept explanations,

					Lecture using videos, Demonstration, PPT	Short summary or overview
	2	Declarations, control structures	6	K4(An)	Lecture using Chalk and talk, Group Discussion, PPT	Problem-solving questions
	3	Functions in C++.	6	K6(C)	Lecture using Chalk and talk, Problem solving	Finish a procedure in many steps, by writing program
IV	Inheritance and overloading					
	1	Classes and Objects	3	K2(U)	Lecture using Chalk and talk, Group Discussion,	Formative Assessment
	2	Constructors and Destructors	3	K6(C)	Lecture using Chalk and talk, Demonstration, PPT	Class test
	3	Operators overloading	3	K4(An)	Lecture using Chalk and talk, Introductory session, PPT	Discussions
	4	Type Conversion	2	K4(An)	Lecture using Chalk and talk, PPT	Class test, online quiz,
	5	Inheritance, Pointers and Arrays	4	K6(C)	Program Demonstration, PPT	Problem-solving questions, home work
V	Memory Management					
	1	Memory Management Operators	3	K4(An)	Group Discussion	Assignments
	2	Polymorphism	3	K4(An)	Lecture using Chalk, Group Discussion, PPT	Map knowledge, questioning
	3	Virtual functions	2	K3(Ap)	Lecture using Chalk and talk, Demonstration, PPT	Suggest idea/concept with examples.
	4	Files, Exception Handling	3	K6(C)	Lecture using Chalk and talk, Demonstration, PPT	Problem-solving questions, Finish a procedure in many steps,

	5	String Handling, Templates	2	K6(C)	Lecture using Chalk and talk, Demonstration, with PPT	Evaluation through writing program. .
VI	1	Expert lectures, online seminars, webinars	2	K2(U)	Lecture, Group Discussion, PPT	Discussions

Course Focussing on Employability/ Entrepreneurship/ Skill Development: Employability

Activities (Em/ En/SD): Asking students to write a program for algorithm.

Assignment: Relationship Among classes, Functions in C++.

Seminar Topic: Functions Declaration

Sample questions

Part A (1 mark)

1. A _____ is a collection of objects of similar type
2. Each class in a program is related with each other (T/F)
3. _____ is an input object in c++..
4. constructor has the name as the class name (T/F)
5. _____ function is used to compare the two strings

Part B (4 marks)

6. Short notes on object Model.
7. Explain about relationships among objects
8. What are control structures? Explain
9. Elaborate about Inheritance
10. Briefly explain about Polymorphism

Part C (8 marks)

11. Explain about Elements of the Object Model.
12. Write about Classification: The importance of Proper Classification
13. Discuss with Functions in C++.
14. Explain about operators overloading with example
15. List out the string handling functions and explain with example.

Head of the Department
Ms. J. Anto Hepzie Bai

Course Instructor
Dr. F. Fanax Femy

Department : Computer Science
Class : I M. Sc Computer Science
Title of the Course : Elective Course I: Python Programming
Semester : I
Course Code : SP231EC1

Course Code	L	T	P	Credits	Inst. Hours	Total Hours	Marks		
							CIA	External	Total
SP231EC1	5	-	-	3	5	75	25	75	100

Objectives

1. Presents an introduction to Python, creation of web applications, network applications and working in the clouds
2. Use functions for structuring Python programs
3. Understand different Data Structures of Python
4. Represent compound data using Python lists, tuples and dictionaries

Course Outcomes

CO	Upon completion of this course, the students will be able to:	PSO addressed	Cognitive level
CO – 1	understand the basic concepts of Python Programming	PSO- 1	K1(R)
CO – 2	understand file operations, Classes and Objects	PSO- 3	K3(Ap)
CO – 3	acquire Object Oriented Skills in Python	PSO- 2	K4(An)
CO – 4	develop web applications using Python	PSO- 4	K6(C)
CO – 5	develop Client Server Networking applications	PSO- 5	K6(C)

Teaching plan

Total Contact hours: 75 (Including lectures, assignments and tests)

Unit	Module	Topic	Teaching Hours	Cognitive level	Pedagogy	Assessment/ Evaluation
I	Introduction					
	1.	Python: Introduction	3	K1(R)	Introductory session	Overview
	2.	Numbers	2	K2(U)	Lecture using Chalk and talk	Simple definitions
	3.	Strings	2	K2(U)	Lecture using Chalk and talk	Solve problems
	4.	Variables	2	K2(U)	PPT	Finish a procedure in many steps
	5.	Lists	2	K2(U)	Lecture using Chalk and talk	Simple programs
	6.	Tuples Dictionaries Sets	2	K2(U)	Demonstration	Concept definitions
	7.	Comparison	2	K4(An)	PPT	Problem solving questions
II	Code Structures					
	1.	Code Structures: if, else if, and else Repeat with while Iterate with for	3	K2(U)	Lecture using Chalk and talk	Evaluation through short test
	2.	Comprehensions	2	K2(U)	Demonstration	Map knowledge
	3.	Functions	2	K2(U)	Lecture using videos	Differentiate between various ideas
	4.	Generators Decorators	2	K4(An)	PPT	Seminar

	5.	Namespaces and Scope	2	K1(R)	Demonstration	Recall steps
	6.	Handle Errors with try and except	2	K5(E)	Demonstration	MCQ
	7.	User Exceptions	2	K1(R)	Lecture using Chalk and talk	Map knowledge
III	Modules, Packages, and Classes					
	1.	Modules, Packages, and Programs: Standalone Programs Command Line Arguments , Modules and the import Statement	3	K1(R)	PPT	Recall steps
	2.	The Python Standard Library	2	K1(R)	Lecture using Chalk and talk	MCQ
	6.	Objects and Classes: Define a Class with class	2	K2(U)	Demonstration	Recall steps
	7.	Inheritance Override a Method Add a Method Get Help from Parent with super	3	K6(C)	Lecture using videos	Short Summary
	8.	In self Defense Get and Set Attribute Values with Properties	2	K3(A)	Demonstration	MCQ
	9.	Name Mangling for Privacy	1	K2(U)	PPT	Concept explanations
	10.	Method Types Duck Typing Special Methods	1	K1(R)	PPT	Assignment
	11.	Composition	1	K6(C)	Demonstration	Quiz

IV Datatypes and Web						
1.	Data Types: Text Strings Binary Data	2	K2(U)	Lecture using Chalk and talk	Short summary	
8.	Storing and Retrieving Data: File Input /Output	2	K1(R)	Lecture using videos	Concept explanations	
9.	Structured Text Files	2	K3(Ap)	Demonstration	Recall Steps	
10.	Structured Binary Files	1	K2(U)	Lecture using Chalk and talk	Concept explanations	
11.	Relational Databases	2	K1(R)	PPT	Evaluation through short test	
12.	No SQL Data Stores	1	K6(C)	Lecture using Chalk and talk	Short summary	
13.	Web: Web Clients Web Servers	2	K1(R)	PPT	Evaluation through short test	
14.	Web Services and Automation	2	K6(C)	Demonstration	Concept explanations	
V Systems and Networks						
8.	Systems: Files Directories	2	K2(U)	Lecture using Chalk and talk	True/False	
9.	Programs and Processes Calendars and Clocks	2	K2(U)	Demonstration	Evaluation through problems	
10.	Networks: Patterns	2	K2(U)	PPT	Recall Steps	
11.	Internet Services	2	K4(An)	Demonstration	MCQ	
12.	The Publish Subscribe	2	K3(A)	PPT	Short essays	

		Model TCP/IP Sockets Zero MQ				
	13.	Concurrency: Queues Processes Threads Green Threads and gevent Twisted Redis	2	K5(E)	Lecture using Chalk and talk	Seminar
	14.	Web Services and APIs Remote Processing Big Fat Data and Map Reduce	2	K1(R)	PPT	Concept explanations
	15.	Working in the Clouds	1	K2(U)	Lecture using videos	Recall
VI		Expert Lectures, Online Seminars, Webinars	2	K2(U)	PPT	Concept explanations

Course Focussing on Employability/ Entrepreneurship/ Skill Development: Skill Development

Activities (Em/ En/SD): 1. Develop programs using Object Oriented Concepts.

2. Creating interactive web pages using forms.

Assignment: Methods in Python and TCP/IP Model

Seminar Topic: Generators and Concurrency Control

Sample questions

Part A (1 Mark)

- ___ is the maximum length of a Python identifier.
(a) 32 b)16 c)128 d) No fixed length
- Which of the following is used for exception handling in Python?
(a) try b) except c) finally d) All the above
- As what data type are the *args stored, when passed into a function?

- (a) List b) Tuple c) Dictionary d) None of the above
4. Which of the following keywords is not reversed keyword in python?
a) Class b) goto c) and d)none
5. Python supports the creation of anonymous functions at run time using a construct called _____.
a) lambda b) pi c) anonymous d) none

Part B (4 Marks)

6. Write the features of Python.
7. Explain why Python is considered as an Interpreted Language.
8. Write notes on Name Mangling for Privacy.
9. Write a Python program to display Fibonacci sequence for n terms.
10. What do you understand about Redis?

Part C (8 Marks)

11. Explain about tuples, lists and dictionaries in Python with example.
12. Explain in detail about Exception Handling.
13. Elaborate Inheritance concepts with examples.
14. Describe Web Services and automation.
15. Write in detail about TCP/IP model.

Head of the Department

Mrs.J.Anto Hepzie Bai

Course Instructor

Dr.S.Immaculate Shyla

Department : **Computer Science**
Class : **I M.Sc. Computer Science**
Title of the Course : **Elective Course II: Advanced Software Engineering**
Semester : **I**
Course Code : **SP231EC4**

Course Code	L	T	P	Credits	Inst. Hours	Total Hours	Marks		
							CIA	External	Total
SP231EC4	5	-	-	3	5	75	25	75	100

Objectives

1. Introduce to Software Engineering, Design, Testing and Maintenance.
2. Enable the students to learn the concepts of Software Engineering.

Course Outcomes

CO	Upon completion of this course, the students will be able to:	PSO addressed	Cognitive level
CO - 1	understand about Software Engineering process	PSO - 1	K1, K2
CO - 2	understand about Software project management skills, design and quality management	PSO - 2	K2, K3
CO - 3	analyze on Software Requirements and Specification	PSO - 3	K3,K4
CO - 4	analyze on Software Testing, Maintenance and Software Re-Engineering	PSO - 4	K4, K5
CO - 5	design and conduct various types and levels of software quality for a software project	PSO - 5	K5, K6

Teaching plan

Total Contact hours: 75 (Including lectures, assignments and tests)

Unit	Module	Topic	Teaching Hours	Cognitive level	Pedagogy	Assessment/ Evaluation
I	Introduction, Software Processes					
	1.	The Problem Domain	2	K2(U)	Lecture using Chalk and talk	Simple explanations, questioning
	2.	Software Engineering Challenges, Software Engineering Approach	3	K1(R)	Lecture using PPT	Recall steps, Concept definitions
	3.	Software Process, Characteristics of a Software Process	4	K4(An)	Lecture cum, Demonstration, PPT	Suggest idea/concept with examples
	4.	Software Development Process Models	2	K3(Ap)	Group Discussion	Finish a procedure in many steps.
	5.	Other software processes	4	K5(E)	Lecture using videos, Demonstration, PPT	Short test
II	Software Requirements Analysis and Specification, Case study					
	1	Requirement engineering, Type of Requirements	2	K4(An)	Lecture using illustrations	Quiz
	2	Feasibility Studies, Requirements Elicitation, Requirement Analysis	2	K2(U)	Lecture Group Discussion,	Evaluation through short test
	3	Requirement Documentation, Requirement Validation,	4	K4(An)	Lecture Discussion, PPT	Finish a procedure in many steps, Map knowledge

		Requirement Management, SRS				
	4	Formal System Specification, Axiomatic Specification, Algebraic Specification	3	K5(E)	Lecture using videos, Demonstration, PPT,	Longer essay/ Evaluation essay, Critique or justify with pros and cons
	5	Student Result management system. Software Quality Management, Software Quality, Software Quality Management System, ISO 9000, SEI CMM.	4	K3(Ap)	Discussion	Seminar.

III Software Project Management

	1	Responsibilities of a software project manager, Project planning	2	K2(U)	Lecture using Chalk and talk, PPT	Evaluation through short test, MCQ,
	2	Metrics for Project size estimation, Project Estimation Techniques, Empirical Estimation Techniques	2	K4(An)	Lecture using PPT	Map knowledge
	3	COCOMO, Halstead' software science, Staffing level estimation	4	K6(C)	Lecture Demonstration, PPT	Class test
	4	Scheduling, Organization and Team Structures, Staffing	4	K5(E)	Lecture using videos, PPT, Demonstration	Discussions, questioning

	5	Risk management, Software Configuration Management, Miscellaneous Plan.	3	K4(An)	Group Discussion.	Quiz
IV	Software Design					
	1	Outcome of a Design process	3	K2(U)	Lecture Demonstration, PPT	Concept explanations, Short summary
	2	Characteristics of a good software design, Cohesion and coupling	3	K6(C)	Lecture with PPT	Differentiate between various ideas
	3	Strategy of Design, Function Oriented Design	3	K4(An)	Group Discussion, PPT	Map knowledge
	4	Object Oriented Design, Detailed Design	2	K4(An)	Lecture using Chalk and talk, PPT	Online quiz
	5	IEEE Recommended Practice for Software Design Descriptions.	4	K6 (C)	Lecture with Demonstration.	Short summary
V	Software Testing					
	1	A Strategic approach to software testing, Terminologies	3	K4(An)	Lecture with PPT	Concept explanations
	2	Functional testing, Structural testing, Levels of testing, Validation testing, Regression testing	3	K4(An)	Peer Teaching	Finish a procedure in many steps, Map knowledge
	3	Art of Debugging, Testing tools, Metrics, Reliability Estimation	2	K3(Ap)	Lecture using Chalk and talk, Demonstration, PPT	Suggest idea/concept with examples,

	4	Software Maintenance, Maintenance Process, Reverse Engineering	3	K5(E)	Lecture using Chalk and talk.	Seminar
	5	Software Reengineering, Configuration Management Activities	2	K6(C)	Group Discussion	Short test
VI	1	Expert lectures, online seminars, webinars	2	K2(U)	Lecture Demonstration, PPT	Concept explanations, Short summary

Course Focussing on Employability/ Entrepreneurship/ Skill Development: Employability, Skill Development

Activities (Em/ En/SD): Making the students to create a data model for one project.

Assignment: Software Process, COCOMO Model

Seminar Topic: Risk Management, Software Maintenance

Sample questions

Part - A (1 mark)

- If the objects focus on the problem domain, then we are concerned with _____.
 - Object Oriented Analysis
 - Object Oriented Design
 - Object Oriented Analysis and Design
 - None of the above
- Software project management comprises of a number of activities, which contains
 - Project planning
 - Scope management
 - Project estimation
 - All mentioned above
- COCOMO stands for _____
- SRS stands for Software Requirement Specification (T/F)
- Software Requirement Specification (SRS) is also known as specification of _____

Part - B (4 marks)

- Short notes on Software Process.
- Explain about Feasibility Studies

8. Discuss about the metrics for Project size Estimation
9. What are the Characteristics of a good software design?
10. Briefly explain about Validation Testing.

Part - C (8 marks)

11. Define Software Development Process Models
12. Explain about Software Quality Management
13. Write about COCOMO Model Techniques.
14. Discuss with Cohesion and Coupling.
15. Explain about different Strategic approach to software testing

Head of the Department

Ms. J. Anto Hepzie Bai

Course Instructor

Dr. F. Fanax Femy