# SANT GADGE BABA AMRAVATI UNIVERSITY GAZETTE

Official Publication of Sant Gadge Baba Amravati University

PART- TWO

**Thursday, the 30th March, 2023**


**NOTIFICATION**

No. 50/2023. Date : 30 /3/2023.

**Subject : Implementation of revised syllabus of M.Sc. Semester- II (Computer Science/Computer Software) under CBCS Pattern.**

It is notified for general information of all the concerned that, the authorities of the University have accepted the revised syllabi of the subject 2MCS1 –Operating System Algorithms & 2 MCS3 – Software Engineering of M.Sc Semester - I I (Computer Science /Computer Software), Which is to be implemented from the Academic Session 2022-2023 under Choice Based Credit System appended herewith as Appendix-A.

Sd/-
(Dr. Tushar Deshmukh)
Registrar,
Sant Gadge Baba Amravati University

**Appendix-A**

**M.Sc. Computer Science/Computer Software**

**Semester-II**

**2MCS1:Operating System Algorithms**

| | |
|---|---|
| **Course Code** | **2MCS1** |
| **Course Name** | **Operating System Algorithms** |
| **Total Credits** | **4** |
| **Course Outcomes** | **CourseOutcome:**<br>**Upon Completion of this course successfully, students would be able to**<br>1. Analyze basic components of Operating Systems and various types of Operating Systems.<br>2. Understand and implement mechanism of OS to handle multiple processes and threads and their communication.<br>3. Classify different mechanisms involved in memory management OS.<br>4. Find the loopholes in the system to resolve the problem of deadlock.<br>5. Study the components and management aspects of concurrency management<br>6. Study and analyze the problems in distributed Operating System. |

| Units | Contents | Total Lectures |
|---|---|---|
| I | Introduction: Services,Types,User-O.S.Interface: Command Interpreter, Graphical User Interface; System Calls; System Programs; Operating System Structure: Simple, Layered Approach; Micro-kernels, Modules; Virtual Machine; System Boot. | 10 |

| | | |
|---|---|---|
| II | Process Management: Process Concept, Process States, Process Control Block, Process Scheduling: Schedulers, Context Switch; Operations on Process: Creation, Termination, Inter Process Communication; Threads: Concept, Benefits; CPU Scheduling: Burst Cycle, Types of Scheduling, Scheduler, Dispatcher, Scheduling Criteria, Scheduling Algorithms: FCFS, SJF, Priority Scheduling, Round- Robin, [multiple processors cheduling] | 10 |
| III | Process Synchronization and Deadlocks: Critical Section Problem, Synchronization Hardware, Semaphores, Classic Problems of Synchronization, Monitors. Deadlock: System Model, Deadlock Characterization, Resource Allocation Graph, Methods for handling Deadlock: Prevention, Avoidance and Detection; Recovery from Deadlock: Process Termination, Resource Preemption. | 10 |
| IV | Memory Management: [Basic Hardware, Address Binding]; Logical and Physical Address Space, Swapping, Contiguous Allocation, Dynamic Storage Allocation: First-fit, Best-Fit, Worst-fit; Fragmentation; Paging; Segmentation.Virtual Memory:Introduction, Virtual Address Space, Demand Paging, Copy-on-write, Page Replacement: Concept, Page Replacement Algorithms: FIFO, Optimal Page Replacement, LRU, Second- Chance Page Replacement; Thrashing,I / OInterlock. | 10 |
| V | FileSystem: File: Concept, Attributes, Operations; FileOrganization and Access: Sequential, Index Sequential, Indexed, Direct or Hash File. Directory: Operations, Structures. Protection: Access Control and Permissions. File System Structure, Allocation Methods, Free Space Management .Disk Structure, Disk Scheduling Algorithms: FCFS, SSTF, SCAN, C-SCAN, LOOK.[Disk Management, Swap Space Management], RAID: Concept. I/O Systems: I/O Hardware, Interrupts, DMA, ApplicationI / OInterface, Kernel I/O Subsystem. | 10 |
| VI | Distributed File System: Concept, Namingand Transparency, Remote File Access, Stateful Vs Stateless Service, File Replication, Remote Login, Remote File Transfer, Data Migration, Computation Migration, Process Migration. Embedded Operating Systems: Embedded Systems: Definition, Requirements and Constraints, Organization of Embedded System; Characteristics of Embedded Operating Systems. Case Studies: Linuxand Mobile Operating Systems<br>CASESTUDY: LINUX and AND ROID Operating System | 10 |
| | **TextBooks:**<br>1. Operating System Concepts –Seventh Edition: Abraham Silberschatz, Peter Galvin, Greg Gagne (John Wiley & Sons)<br>2. Operating Systems:William Stallings (Pearson)<br>3. Modern Operating System :Andrew S. Tanenbaum | |
| | **ReferenceBooks:**<br>1. Modern Operating Systems, Pearson Education A.S.Tanenbaum<br>2. Advanced Conceptsin Operating SystemM. Singhal & N. Shivaratri<br>3. Distributed Operating System byA. S.Tanenbaum<br>4. Linux Kernel, 2nd Edition By Daniel P. Bovet,O' Reilly<br>5. The Design of Unix Operating System Maurice Bach, Pearson | |

**M.Sc. Computer Science / Computer Software**
**Semester-II**
**2MCS3:Software Engineering**

| | |
|---|---|
| **Course Code** | **2MCS3** |
| **Course Name** | **3Software Engineering** |
| **Total Credits** | **4** |
| **Course Outcomes** | **Course Outcome:**<br>**Upon completion of this course successfully, students would be able to**<br>1. Identify the need for engineering approach for software development and various processes of requirements, analysis for software engineering problems.<br>2. Analyze various software engineering models and apply methods for design and development of software projects.<br>3. Apply various techniques, metrics and strategies for Testing software projects.<br>4. Identify and apply the principles, processes and main knowledge are as for Software Project Management<br>5. Proficiently apply standards, CASE tools and techniques for Engineering software projects |

| Units | Contents | Total Lectures |
|---|---|---|
| I | The Software Process, Software Engineering, Generic process model, Prescriptive Process Models: The Waterfall Model, Incremental Process Models, Evolutionary Process Models, Concurrent Models, A Final Word on Evolutionary Processes. Agile Development: What is an Agile Process?, Agility Principles, The Politics of Agile Development, Human Factors. Extreme Programming (XP): XP Values, The XP Process, Industrial XP, The XP Debate. Requirements Modeling: Scenarios, Information And Analysis Classes: Requirements Analysis: Overall Objectives and Philosophy, Analysis Rules of Thumb, Domain Analysis, Requirements Modelling Approaches, Scenario-Based Modelling: Creating a Preliminary Use Case, Refining a Preliminary Use Case, Writing a Formal Use Case | 10 |
| II | Design Concepts: Design within the Context of Software Engineering, TheDesign Process: Software Quality Guidelines and Attributes, The Evolution of Software Design, Design Concepts: Abstraction, Architecture, Patterns,Separation of Concerns, Modularity, Information Hiding, Functional Independence, Refinement, Aspects, Refactoring, Object-Oriented Design Concepts, Design Classes, Architectural Design: Software Architecture:Whatis Architecture?, Why Is Architecture Important?,Architectural Descriptions, Architectural Decisions, Component- Level Design: What Is a Component? An Object-Oriented View, The Traditional View, A Process RelatedView, Designing Class-Based Components, Basic Design Principles, Component-Level Design Guidelines, Cohesion, Coupling, User Interface Design: The Golden Rules: Place the User in Control, Reduce the User's Memory Load, Make the Interface Consistent. | 10 |
| III | Quality Management: Quality Concepts: What Is Quality? Software Quality, Garvin's Quality Dimensions, McCall's Quality Factors, Targeted QualityFactors, The transition to a Quantitative View, Review Techniques: FormalTechnical Reviews. Software Quality Assurance,: Elements of Software Quality Assurance ,SQATasks,Goals and Metrics: SQA Tasks, Goals, Attributes, and Metrics,Formal Approachesto SQA: Statistical Software Quality Assurance: A Generic Example, SixSigma for Software Engineering, Software Reliability: Measures of Reliability and Availability, Software Safety. | 10 |

| | | |
|---|---|---|
| **IV** | Software Configuration Management: An SCM Scenario, Elements of a Configuration Management System, Baselines, Software Configuration Items, The SCM Repository: The Role of the Repository, General Featuresand Content, SCM Features, The SCM Process: Identification of Objects inthe Software Configuration, Version Control, Change Control, ConfigurationAudit, Status Reporting. Product Metrics: A Framework for Product Metrics: Measures, Metrics, and Indicators, The Challenge of Product Metrics, Measurement Principles, Goal-Oriented Software Measurement, The Attributes of Effective Software Metrics, Metrics for the Requirements Model: Function-Based Metrics, Metrics for Specification Quality, Metrics for the Design Model: Architectural Design Metrics, Metrics for Object-Oriented Design,Class-Oriented Metrics-Class-Oriented Metrics, OO Metrics Proposed by Lorenz and Kidd, Component Level Design Metrics, Operation-Oriented Metrics, User Interface Design Metrics | 10 |
| **V** | Managing Software Projects: Project Management Concepts:The Management Spectrum: The People, The Product, The Process, The Project, The Product: Software Scope,Problem Decomposition, TheProcess: Melding the Product and the Process, Process Decomposition, Process And Project Metrics: Metrics in the Process and Project Domains: Process Metrics and Software Process Improvement, Project Metrics, Software Measurement: Size-Oriented Metrics, Function Oriented Metrics, Reconciling LOC and FPMetrics, Object-Oriented Metrics, Use Case-Oriented Metrics. Metrics for Software Quality: Measuring Quality, Defect Removal Efficiency. Resources: Human Resources, Reusable Software Resources, Environmental Resources, Software Project Estimation, DecompositionTechniques: Software Sizing, Problem- Based Estimation, An Example of LOC-Based Estimation, An Example of FP-Based Estimation, Process-Based Estimation, An Example of Process-Based Estimation, Estimation with Use Cases, An Example of Use Case Based Estimation, Reconciling Estimates | 10 |
| **VI** | Project Scheduling: Basic Concepts, Project Scheduling: Basic Principles,The Relationship Between People and Effort, Effort Distribution, Defining aTaskSetfortheSoftware Project:ATask Set Example, Refinement of Software Engineering Actions, Scheduling: Time-line Charts, Tracking the Schedule, Tracking Progress for an OO Project, Scheduling for Web App Projects, Risk Management: Reactive versus Proactive Risk Strategies, Software Risks, Risk Identification: Assessing Overall Project Risk, Components and Drivers, Risk Projection: Developing a RiskTable, Assessing Risk Impact, Risk Refinement, RMMM plan | 10 |

**TextBooks:**

1.Software Engineering:A Practitioner's Approach. Roger S. Pressman, Seventh edition McGraw Hill International Edition.

**ReferenceBooks:**

1. Software Engineering, A Precise Approach, Pankaj Jalote,Wiley India,2010.
2. Software Engineering: APrimer, Waman S Jawadekar, Tata McGraw-Hill,2008
3. Fundamental sofSoftware Engineering, RajibMall,PHI,2005
4. Software Engineering, Principles and Practices, Deepak Jain, Oxford University Press.
5. Software Engineering 1:Abstract ionandmodeling, Diner Bjorner, Springer International edition,2006.
6. Software Engineering 2: Specificati on of systems and languages, Diner Bjorner,

******