

SOFTWARE TESTING [As per Choice Based Credit System (CBCS) scheme] (Effective from the academic year 2016 -2017) SEMESTER – V			
Subject Code	15IS63	IA Marks	20
Number of Lecture Hours/Week	4	Exam Marks	80
Total Number of Lecture Hours	50	Exam Hours	03
CREDITS – 04			
Course objectives: This course will enable students to <ul style="list-style-type: none"> • Differentiate the various testing techniques • Analyze the problem and derive suitable test cases. • Apply suitable technique for designing of flow graph • Explain the need for planning and monitoring a process 			
Module – 1			Teaching Hours
Basics of Software Testing: Basic definitions, Software Quality , Requirements, Behaviour and Correctness, Correctness versus Reliability, Testing and Debugging, Test cases, Insights from a Venn diagram, Identifying test cases, Test-generation Strategies, Test Metrics, Error and fault taxonomies , Levels of testing, Testing and Verification, Static Testing. Problem Statements: Generalized pseudocode, the triangle problem, the NextDate function, the commission problem, the SATM (Simple Automatic Teller Machine) problem, the currency converter, Saturn windshield wiper T1:Chapter1, T3:Chapter1, T1:Chapter2.			10 Hours
Module – 2			10 Hours
Functional Testing: Boundary value analysis, Robustness testing, Worst-case testing, Robust Worst testing for triangle problem, Nextdate problem and commission problem, Equivalence classes, Equivalence test cases for the triangle problem, NextDate function, and the commission problem, Guidelines and observations, Decision tables, Test cases for the triangle problem, NextDate function, and the commission problem, Guidelines and observations. Fault Based Testing: Overview, Assumptions in fault based testing, Mutation analysis, Fault-based adequacy criteria, Variations on mutation analysis. T1: Chapter 5, 6 & 7, T2: Chapter 16			
Module – 3			10 Hours
Structural Testing: Overview, Statement testing, Branch testing, Condition testing , Path testing: DD paths, Test coverage metrics, Basis path testing, guidelines and observations, Data –Flow testing: Definition-Use testing, Slice-based testing, Guidelines and observations. Test Execution: Overview of test execution, from test case specification to test cases, Scaffolding, Generic versus specific scaffolding, Test oracles, Self-checks as oracles, Capture and replay T3:Section 6.2.1, T3:Section 6.2.4, T1:Chapter 9 & 10, T2:Chapter 17			
Module – 4			10 Hours
Process Framework : Basic principles: Sensitivity, redundancy, restriction, partition, visibility, Feedback, the quality process, Planning and monitoring, Quality goals, Dependability properties ,Analysis Testing, Improving the process, Organizational factors. Planning and Monitoring the Process: Quality and process, Test and analysis strategies and plans, Risk planning, monitoring the process, Improving the			

<p>process, the quality team</p> <p>Documenting Analysis and Test: Organizing documents, Test strategy document, Analysis and test plan, Test design specifications documents, Test and analysis reports.</p> <p>T2: Chapter 3 & 4, T2: Chapter 20, T2: Chapter 24.</p>	
<p>Module – 5</p>	
<p>Integration and Component-Based Software Testing: Overview, Integration testing strategies, Testing components and assemblies. System, Acceptance and Regression Testing: Overview, System testing, Acceptance testing, Usability, Regression testing, Regression test selection techniques, Test case prioritization and selective execution. Levels of Testing, Integration Testing: Traditional view of testing levels, Alternative life-cycle models, The SATM system, Separating integration and system testing, A closer look at the SATM system, Decomposition-based, call graph-based, Path-based integrations.</p> <p>T2: Chapter 21 & 22, T1 : Chapter 12 & 13</p>	<p>10 Hours</p>
<p>Course outcomes: The students should be able to:</p> <ul style="list-style-type: none"> • Derive test cases for any given problem • Compare the different testing techniques • Classify the problem into suitable testing model • Apply the appropriate technique for the design of flow graph. • Create appropriate document for the software artefact. 	
<p>Question paper pattern: The question paper will have TEN questions. There will be TWO questions from each module. Each question will have questions covering all the topics under a module. The students will have to answer FIVE full questions, selecting ONE full question from each module.</p>	
<p>Text Books:</p> <ol style="list-style-type: none"> 1. Paul C. Jorgensen: Software Testing, A Craftsman's Approach, 3rd Edition, Auerbach Publications, 2008. (Listed topics only from Chapters 1, 2, 5, 6, 7, 9, 10, 12, 13) 2. Mauro Pezze, Michal Young: Software Testing and Analysis – Process, Principles and Techniques, Wiley India, 2009. (Listed topics only from Chapters 3, 4, 16, 17, 20, 21, 22, 24) 3. Aditya P Mathur: Foundations of Software Testing, Pearson Education, 2008. (Listed topics only from Section 1.2 , 1.3, 1.4 ,1.5, 1.8,1.12,6. 2.1,6. 2.4) 	
<p>Reference Books:</p> <ol style="list-style-type: none"> 1. Software testing Principles and Practices – Gopalaswamy Ramesh, Srinivasan Desikan, 2nd Edition, Pearson, 2007. 2. Software Testing – Ron Patton, 2nd edition, Pearson Education, 2004. 3. The Craft of Software Testing – Brian Marrick, Pearson Education, 1995. 4. Anirban Basu, Software Quality Assurance, Testing and Metrics, PHI, 2015. 5. Naresh Chauhan, Software Testing, Oxford University press. 	

gchh

H.O.D.

Dept. Of Information Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOGOBIDRI - 574 225

structure; Disk attachment; Disk scheduling; Disk management; Swap space management. Protection: Goals of protection, Principles of protection, Domain of protection, Access matrix, Implementation of access matrix, Access control, Revocation of access rights, Capability- Based systems. Case Study: The Linux Operating System: Linux history; Design principles; Kernel modules; Process management; Scheduling; Memory Management; File systems, Input and output; Inter-process communication.	
Course outcomes: The students should be able to:	
<ul style="list-style-type: none"> • Demonstrate need for OS and different types of OS • Apply suitable techniques for management of different resources • Use processor, memory, storage and file system commands • Realize the different concepts of OS in platform of usage through case studies 	
Question paper pattern: The question paper will have TEN questions. There will be TWO questions from each module. Each question will have questions covering all the topics under a module. The students will have to answer FIVE full questions, selecting ONE full question from each module.	
Text Books:	
1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, Operating System Principles 7 th edition, Wiley-India, 2006.	
Reference Books	
1. Ann McHoes Ida M Fylnn, Understanding Operating System, Cengage Learning, 6 th Edition 2. D.M Dhamdhare, Operating Systems: A Concept Based Approach 3rd Ed, McGraw-Hill, 2013. 3. P.C.P. Bhatt, An Introduction to Operating Systems: Concepts and Practice 4th Edition, PHI(EEE), 2014. 4. William Stallings Operating Systems: Internals and Design Principles, 6th Edition, Pearson.	

Signature

H.O.D.

Dept. Of Information Science & Engineering
 Alva's Institute of Engg. & Technology
 Mijar, MOODBIDRI - 574 225