Code No. MCA 110	**L T C**
Paper: Software Engineering	**3 1 4**

INSTRUCTIONS TO PAPER SETTERS:

1. Question No. 1 should be compulsory and cover the entire syllabus. There should be 10 questions of short answer type of 2 marks each, having at least 2 questions from each unit.
2. Apart from Question No. 1, rest of the paper shall consist of four units as per the syllabus. Every unit should have two questions to evaluate analytical/technical skills of candidate. However, student may be asked to attempt only 1 question from each unit. Each question should be 10 marks including subparts, if any.

OBJECTIVE: *Course is intended to help students to develop skills that will enable them to construct software of high quality – software that is reliable, and that is reasonably easy to understand, modify and maintain. The concept covered in syllabus are*

- *The software development process.*
- *Software requirements and specifications.*
- *Software design techniques.*
- *Techniques for developing large software systems.*
- *CASE tools and software development environments.*
- *Software testing, documentation and maintenance.*

PRE-REQUISITE:

- Program Development
- Basic Concepts of Data Management

UNIT-I

Introduction: Software Crisis, Software Processes & Characteristics, Software life cycle models, Waterfall, Prototype, Evolutionary and Spiral Models.

Software Requirements analysis & specifications: Requirement engineering, requirement elicitation techniques like FAST, QFD & Use case approach, requirements analysis using DFD, Data dictionaries & ER Diagrams, Requirements documentation, Nature of SRS, Characteristics & organization of SRS, Requirement Management, IEEE Std. for SRS. **[No. of Hrs.: 10]**

UNIT-II

Software Project Planning: Size Estimation like lines of Code & Function Count, Cost Estimation Models, COCOMO, Putnam resource allocation model, Validating Software Estimates, Risk Management.

Software Design: Cohesion & Coupling, Classification of Cohesiveness & Coupling, Function Oriented Design, Object Oriented Design. **[No. of Hrs.: 12]**

UNIT-III

Software Metrics: Software measurements: What & Why, Token Count, Halstead Software Science Measures, Data Structure Metrics, Information Flow Metrics.

Software Reliability: Importance, Hardware Reliability & Software Reliability, Failure and Faults, Reliability Models- Basic Model, Logarithmic Poisson Model, Software Quality Models, CMM & ISO 9001. **[No. of Hrs.: 10]**

UNIT-IV

Software Testing: Testing process, Design of test cases, Introduction to functional testing & Structural testing, Unit Testing, Integration and System Testing, Debugging, Alpha & Beta Testing.

Software Maintenance: Management of Maintenance, Maintenance Process, Maintenance Models, Regression Testing, Reverse Engineering, Software Re-engineering, Configuration Management, Documentation. **[No. of Hrs.: 10]**

TEXT BOOKS:

1. K. K. Aggarwal and Yogesh Singh, "Software Engineering", New Age International, 3rd Ed., 2005.
2. R. S. Pressman, "Software Engineering – A Practitioner's Approach", McGraw Hill Int. , 5th Ed., 2001.
3. Pankaj Jalote, "An Integrated Approach to Software Engineering", Narosa, 3rd Ed., 2005.

REFERENCES:

1. Stephen R. Schach, "Classical & Object Oriented Software Engineering", IRWIN, 1996.
2. James Peter, W. Pedrycz, "Software Engineering: An Engineering Approach", John Wiley & Sons.
3. I. Sommerville, "Software Engineering", Addison Wesley,8th Ed., 2009.
4. Frank Tsui and Orlando Karan, "Essentials of Software Engineering", Joes and Bartlett, 2nd Ed., 2010.
5. Kassem A. Saleh, "Software Engineering", Cengage Learning, 2009.
6. Rajib Mall, "Fundamrntal of Software Engineering", PHI, 3rd Ed., 2009.
7. Carlo Ghizzi , Mehdi Jazayeri and Dino Mandrioli, " Fundamental of Software Engineering", PHI, 2nd Ed.,2003.
8. Carol L. Hoover, Mel Rosso-Llopart and Gil Taran, "Evaluating Project Decision Case Studies in Software Engineering", Pearson, 2010.