

Second Semester (Total Marks: 600, Total Credit:20)

Theoretical Courses

Paper-V

CSE 101E TH: Elective Papers

Credit:4

CSE 101E1: Digital Forensics

Course Outcomes (CO):

- i. Analyse and evaluate the importance of personal data its privacy and security.
- ii. Analyse and evaluate the security aspects of social media platforms and ethical aspects associated with use of social media.
- iii. Evaluate and communicate the human role in security systems with an emphasis on ethics, social engineering vulnerabilities and training.
- iv. Increase awareness about cyber-attack vectors and safety against cyber-frauds.
- v. Take measures for self-cyber-protection as well as societal cyber-protection.

Syllabus:

Digital Forensics - Introduction to Digital Forensics; Acquiring Evidence; File systems; Open-Source Forensic Tools; Windows / Mac / Linux Forensics; Advanced Windows Forensics; Programming for Digital Forensics; Application & Database Forensics; Network Forensics; Volatile Memory Analysis; Malware Analysis; Threat Hunting & Incident Response.

References:

1. Cyber Crime Impact in the New Millennium, by R. C Mishra, Auther Press. Edition 2010.
2. Cyber Security Understanding Cyber Crimes, Computer Forensics and Legal Perspectives by Sumit Belapure and Nina Godbole, Wiley India Pvt. Ltd. (First Edition, 2011)
3. Cyber Laws: Intellectual Property & E-Commerce Security by Kumar K, Dominant Publishers.
4. Fundamentals of Network Security by E. Maiwald, McGraw Hill.
5. Digital Evidence and Computer Crime: Forensic Science, Computers, and the Internet, 3rd Edition Author: Eoghan Casey Publisher: Academic Press ISBN: 9780123742681
6. The Art of Memory Forensics Authors: Michael Hale Ligh, Andrew Case, Jamie Levy, Aaron Walters Publisher: Wiley ISBN: 1118825098
7. File System Forensic Analysis (1st Ed.) Author: Brian Carrier Publisher: Addison-Wesley, 2005 ISBN: 0321268172

CSE 101 E2: Software Engineering

Introduction and Brief Overview - Software process, modeling and analysis, software architecture, software design. **Software Modeling, Analysis, Testing** - Analysis modeling and best practices, traditional practice diagrams such as DFDs and ERDs etc, **Traditional Testing techniques** – white box and black box testing. **Object-Oriented Software Engineering** - **Concept of OO Software** – Design and Analysis, Overview of various UML diagrams and UML analysis modeling, analysis case studies, analysis tools, analysis patterns, OO software testing. **Case study with complete examples Software Architecture** - Architectural styles, architectural patterns, analysis of architectures, formal descriptions of software architectures, architectural description languages and tools, scalability and interoperability issues, web

application architectures, case studies. **Software Design** - Design best practices, design patterns, extreme programming, refactoring, design case studies, component technology, object oriented frameworks, distributed objects, object request brokers, case studies. Web Engineering, Clean room Engineering and other recent topics.

References:

1. G. Booch, J. Rumbaugh, and I. Jacobson, I. The Unified Modeling Language User Guide. Addison-Wesley, 1999
 2. E. Gamma, R. Helm, R. Johnson, and J. Vlissides. Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley, 1995
 3. M. Shaw and D. Garlan. Software Architecture: Perspectives on an Emerging Discipline. Prentice-Hall, 1996
 4. L. Bass, P. Clements, and R. Kazman. Software Architecture in Practice, Addison-Wesley, 1998
 5. J. Rumbaugh, I. Jacobson, and G.Booch. The Unified Modeling Language Reference Manual. Addison Wesley Longman, 1999.
 6. Jacobson, G. Booch, and J. Rumbaugh, and I. Jacobson. The Unified Software Development Process. Addison Wesley Longman, 1999.
 7. J. Rumbaugh, M. Blaha, W. Premerlani, F. Eddy, and W. Lorenson. Object-oriented Modeling and Design. PHI, EEE, 1997.
 8. G. Booch. Object-Oriented Analysis and Design with Applications. Second Edition. Benjamin Cummings, 1994.
 9. Jim Conallen. Building Web Applications with UML. Addison-Wesley, 2000.
 10. K. Beck. Extreme Programming Explained. Pearson Education Asia, 2000.
 11. Software engineering – design, reliability and management – Schuman Mar.
 12. Software engineering – Pressman.
-