| **Format No.:LP01**  **Issue No.:01**  **Issue Date:07/01/09** |
| --- |

**SRM INSTITUTE OF SCIENCE AND TECHNOLOGY**

**RAMAPURAM CAMPUS**

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

**ACADEMIC YEAR 2021-22 (ODD SEMESTER)**

**LESSON PLAN**

**Degree/Branch: B.Tech / Cyber Security Total No. of Hours given in syllabus: 75**

**Year/Sem/Sec: II / III /B Tutorial : NIL**

**Subject Code :18CSC202J Lecture : 45**

**Subject Name: Object Oriented Design and Programming Practical : 30**

**Faculty Name: Mrs.S.MENAKA Grand Total : 75**

| **Session** | **Description of Topic** | **Lecture Hours** | **Reference Books** | **Date Planned** | **Date Actually Taken** | **Teaching Methodology** |
| --- | --- | --- | --- | --- | --- | --- |
| **UNIT-1** | | | | | | |
|  | Comparison of Procedural and Object Oriented Programming, OOPS and its features | 1 | LR2,5 |  |  | Inquiry Based Learning |
|  | I/O Operations, Data Types, Variables, static, Constants, Pointers, Type Conversions | 1 | LR2,5 |  |  | PPT |
|  | Features: Class and Objects, UML Diagrams Introduction | 1 | LR1 |  |  | PPT |
|  | Feature: Class and Objects, Examples of Class and Objects | 1 | LR1 |  |  | Flipped Classroom |
|  | UML Class Diagram and its components, Class Diagram relations and Multiplicity | 1 | LR1,7 |  |  | PPT |
|  | Feature: Abstraction and Encapsulation, Application of Abstraction and Encapsulation | 1 | LR1,2 |  |  | PPT |
|  | Access specifiers – public, private,  Access specifiers - protected, friend, inline | 1 | LR2,5 |  |  | Inquiry Based Learning |
|  | UML use case Diagram, use case, Scenario, Use case Diagram objects and relations | 1 | LR1,7 |  |  | Collaborative learning |
|  | Method, Constructor and Destructor, Example program for constructor | 1 | LR2,5 |  |  | PPT |
| **CLA-1** | | | | **Assignment-1**  (Use case and Class diagram for selected Project) | | |
| **UNIT-II** | | | | | | |
|  | Types of constructor (Default, Parameter), Static and copy constructor | 1 | LR 3,5 |  |  | PPT |
|  | Feature Polymorphism: Constructor overloading, Method Overloading | 1 | LR 5 |  |  | Flipped Classroom |
|  | Example for method overloading, Method Overloading: Different parameter with different return values | 1 | LR 5 |  |  | PPT |
|  | Operator Overloading and types, Overloading Assignment Operator | 1 | LR 5 |  |  | Inquiry Based Learning |
|  | Overloading Unary Operators, Example for Unary Operator overloading | 1 | LR 5 |  |  | PPT |
|  | Overloading Binary Operators, Example for Binary Operator overloading | 1 | LR 5 |  |  | PPT |
|  | UML Interaction Diagrams, Sequence Diagram | 1 | LR 1,7 |  |  | Project-Based Learning |
|  | Collaboration Diagram, Example Diagram | 1 | LR 1,7 |  |  | Collaborative learning |
|  | Feature: Inheritance, Inheritance and its types | 1 | LR 5 |  |  | PPT |
| **UNIT-III** | | | | | | |
| 19. | Feature Inheritance: Single and Multiple, Inheritance: Multilevel | 1 | LR 5 |  |  | PPT |
| 20. | Inheritance: Hierarchical, Inheritance: Hybrid | 1 | LR 5 |  |  | PPT |
| 21. | Inheritance: Example Programs | 1 | LR 5 |  |  | Inquiry Based Learning |
| 22. | Advanced Functions: Inline, Friend, Advanced Functions: Virtual, Overriding | 1 | LR 5 |  |  | PPTn |
| 23. | Advanced Function: Pure Virtual function, Example for Virtual and pure virtual function | 1 | LR 5 |  |  | Flipped Classroom |
| 24. | Abstract class and Interface, Example Program | 1 | LR 5 |  |  | PPT |
| 25. | UML State Chart Diagrams, | 1 | LR 1,7 |  |  | PPT |
| 26. | Example State Chart Diagram, UML Activity Diagram | 1 | LR 1,7 |  |  | Collaborative learning |
| 27. | UML Activity Diagram, Example Activity Diagram | 1 | LR 1,7 |  |  | Crossover Learning |
| **CLA-2** | | | | **Assignment-2**  (Sequence diagram, Activity and State chart diagram for the selected Projects) | | |
| **UNIT-1V** | | | | | | |
| 28. | Generic – Templates, Introduction, Function templates | 1 | LR 5 |  |  | PPT |
| 29. | Example programs Function templates, Class Templates | 1 | LR 5 |  |  | PPT |
| 30. | Class Templates, Example programs for Class and Function templates  , | 1 | LR 5 |  |  | Mind Mapping |
| 31. | Exceptional Handling: try and catch, Exceptional Handling: Multilevel exceptional | 1 | LR 3,5 |  |  | PPT |
| 32. | Exceptional Handling: throw and throws, Exceptional Handling: finally | 1 | LR 3,5 |  |  | PPT |
| 33. | Exceptional Handling: User defined exceptional, Example Programs using C++ | 1 | LR 3,5 |  |  | PPT |
| 34. | Dynamic Modeling: Package Diagram ,UML Component Diagram | 1 | LR 1,7 |  |  | Collaborative learning |
| 35. | UML Component Diagram, UML Deployment Diagram | 1 | LR 1,7 |  |  | PPT |
| 36. | UML Deployment Diagram,  Example Package, Deployment, Package | 1 | LR 1,7 |  |  | Crossover Learning |
| **UNIT-V** | | | | | | |
| 37. | STL: Containers: Sequence and Associative Container | 1 | LR 5 |  |  | PPT |
| 38. | Sequence Container: Vector, List, Sequence Container: Deque, Array | 1 | LR 5 |  |  | PPT |
| 39. | STL : Stack | 1 | LR 5 |  |  | PPT & Discussion Forum |
| 40. | Associative Containers: Map, Multimap | 1 | LR 5 |  |  | PPT |
| 41. | Iterator and Specialized iterator, Functions of iterator | 1 | LR 5 |  |  | PPT |
| 42. | Algorithms: find(), count(), sort(), Algorithms: search(), merge() | 1 | LR 5 |  |  | Demo Session |
| 43. | Function Object : for\_each(), transform(), Example for Algorithms | 1 | LR 5 |  |  | PPT |
| 44. | Streams and Files: Introduction, Classes and Errors | 1 | LR 5 |  |  | PPT |
| 45. | Disk File Handling Reading Data and Writing Data | 1 | LR 5 |  |  | Constructivist |
| **CLA-3** | | | | **Assignment-3**  (Component diagram, package diagram and detailed project presentation) | | |

| **Learning Assessment** | | | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Bloom’s level of Thinking** | **Continuous Learning Assessment (50% weightage)** | | | | | | | | **Final Examination (50% Weightage)** | |
| CLA-1(10) | | CLA-2(15%) | | CLA-3(15%) | | CLA-4(10%) | |
| Theory | Practical | Theory | Practical | Theory | Practical | Theory | Practical | Theory | Practical |
| **Level 1** | **Remember** | 20% | 20% | 15% | 15% | 15% | 15% | 15% | 15% | 15% | 15% |
| **Understand** |
| **Level 2** | **Apply** | 20% | 20% | 20% | 20% | 20% | 20% | 20% | 20% | 20% | 20% |
| **Analyze** |
| **Level 3** | **Evaluate** | 10% | 10% | 15% | 15% | 15% | 15% | 15% | 15% | 15% | 15% |
| **Create** |
|  | **Total** | 100% | | 100% | | 100% | | 100% | | 100% | |

| **LEARNING RESOURCES** | |
| --- | --- |
| **S.No** | **TEXT BOOKS** |
|  | Grady Booch, Robert A. Maksimchuk, Michael W. Engle, Object-Oriented Analysis and Design with Applications, 3rd ed., Addison-Wesley, May 2007 |
|  | ReemaThareja, Object Oriented Programming with C++, 1st ed., Oxford University Press, 2015 |
|  | SouravSahay, Object Oriented Programming with C++, 2nd ed., Oxford University Press, 2017 |
|  | Grady Booch, Robert A. Maksimchuk, Michael W. Engle, Object-Oriented Analysis and Design with Applications, 3rd ed., Addison-Wesley, May 2007 |
|  | Robert Lafore, Object-Oriented Programming in C++, 4th ed., SAMS Publishing, 2008 |
|  | AliBahrami, Object Oriented Systems Development”, McGraw Hill, 2004 |
|  | Craig Larmen, Applying UML and Patterns, 3rd ed., Prentice Hall, 2004 |

**SUBJECT HANDLER COURSE COORDINATOR HOD/CSE**