

# DEPARTMENT OF IT

# COURSE PLAN

|  |  |  |  |
| --- | --- | --- | --- |
| Academic Year: | 2021-22 |  |  |
| Subject: | Object Oriented Programming through Java Lab | Faculty Name: | N.Sreevidya |
| Year & Semester: | II year- I Semester | Branch: | IT |

|  |  |  |
| --- | --- | --- |
| **S.No.** | **EXPERIMENT NAME** | **No. of**  **Periods** |
|  | **CYCLE-1** |  |
| 1. | A) Write a program to print prime numbers up to a given number.  B) Write a program to print roots of a quadratic equation ax2+bx+c=0.  C) Write a program to print Fibonacci sequence up to a given number. | 3 |
| 2. | Define a class to represent a bank account and include the following members. **Instance variables:**  (i)Name of depositor  (ii)Account No  (iii)Type of account  (iv)Balance amount in the account  **Instance Methods:**  To assign instance variables (Constructors-Zero argument and parameterized)   1. To deposit an amount 2. To withdraw amount after checking the balance   (iv) To display name and address  Define ExecuteAccount class in which define main method to test above class.  B) In the above account class, maintain the total no. of account holders present in the bank and also define a method to display it. Change the main method appropriately.  C) In main method of ExecuteAccount class, define an array to handle five accounts.  D) In Account class constructor, demonstrate the use of “this” keyword.  E) Modify the constructor to read data from keyboard.  F) Overload the method deposit() method (one with argument and another without argument)  G) In Account class, define set and get methods for each instance variable. | 3 |
| 3. | A) Define Resister class in which we define the following members:  Instance variables:  resistance  Instance Methods:  giveData():To assign data to the resistance variable  displayData(): To display data in the resistance variable  constructors  Define subclasses for the Resistor class called SeriesCircuit and ParallelCircuit in which define methods : calculateSeriesResistance( ) and calculateParallelResistance() respectively.Both the methods should take two Resistor objects as arguments and return Resistor object as result.In main method , define another class called ResistorExecute to test the above class.  B) Modify the above two methods which should accept array of Resistor objects as argument and return Resistor object as result.  C) Write a program to demonstrate method overriding.  D) Write a program to demonstrate the uses of “super” keyword (three uses)  E) Write a program to demonstrate dynamic method dispatch (i.e .Dynamic polymorphism). | 3 |
| 4. | A) Write a program to check whether the given string is palindrome or not.  B) Write a program for sorting a given list of names in ascending order.  C) Write a program to count the no. of words in a given text.  D) Define an interface “GeomtricShape” with methods area( ) and perimeter( ) (Both method’s return type and parameter list should be void and empty respectively.  Define classes like Triangle, Rectangle and Circle implementing the “GeometricShape”  interface and also define “ExecuteMain” class in which include main method to test the above class. | 3 |
| 5. | a) Define a package with name “sortapp” in which declare an interface “SortInterface” with method sort( ) whose return type and parameter list should be void and empty.Define “subsortapp” as subpackage of “sortapp” package in which define class “SortImpl” implementing “SortInterface” in which sort() method should print a message linear sort is used.  Define a package “searchingapp” in which declare an interface “SearchInterface” with search( ) method whose return type and parameter list should be void and empty respectively.  Define “searchingimpl” package in which define a “SearchImpl” class implementing “SearchInterface” defined in “searchingapp” package in which define a search( ) method which should print a message linear search is used.  Define a class ExecutePackage with main method using the above packages(classes and its methods).  Use ArrayList class of Collections Framework to and use algorithms to search and sort the elemnt of an array. | 3 |
| 6. | Modify the withdraw() method of Account class such that this method should throw “InsufficientFundException” if the account holder tries to withdraw an amount that leads to condition where current balance becomes less than minimum balance otherwise allow the account holder to withdraw and update the balance accordingly. | 3 |
| 6. | Revision | 3 |
| 7. | Internal exam-1 | 3 |
|  | **CYCLE-2** |  |
| 1. | A) Define two threads such that one thread should print even numbers and another thread should print odd numbers.  B) Modify the Account class to implement thread synchronization concept.  C) Define two threads such that one thread should read a line of text from text file and another thread should write that line of text to another file. (Thread communication example).  D) Write a program to implement thread priority. | 3 |
| 2 | Design the user screen as follows and handle the events appropriately.  Add Window  First Number  Second Number  Result  ADD  SUBTRACT | 3 |
| 3 | Write a program to simulate a calculator | 3 |
| 4 | a) Write a Java program for handling mouse events and key events.  b) Write a program for handling window events.  c) ) Develop an applet that displays a simple message. | 3 |
| 5 | a) Develop a client that sends data to the server and also develop a server that sends data to the client (two way communication)  b) Develop a client/server application in which client read a file name from keyboard and send the file name to the server, and server will read the file name from client and send the file contents to the client. | 3 |
| 6. | Extra Experiments | 3 |
|  | 1. Write a program to illustrate the scope of default , private, protected and public access modifiers for variables and methods in java . | 3 |
|  | 1. Write a java program to illustrate the usage of multiple catch blocks in java. 2. Develop a program for passing parameters to an Applet. | 3 |
| 7. | Revision | 3 |
| 8. | Internal exam-2 | 3 |
|  | **TOTAL** | 54 |

|  |  |  |
| --- | --- | --- |
| **Faculty** | **Head of Department** | **Principal** |