Object Oriented Programming Lab (PCC-CS593)

/* Programming problems marked with [*] beside should only be written in the Laboratory Note Book; the others are treated as practice problems. But try to remember that every programming problem mentioned here is important in your course. */

Assignment-1 (one day)

Objective:
*The objective of this assignment is to learn how to write some **Simple JAVA Programs.***

1. Write a program to print your name.
2. Write a program to read the price of an item in the decimal form (like 75.95) and print the output in paise (like 7595 paise).
3. Write a program to convert the given temperature in *Fahrenheit* to *Celsius* using the following conversion formula:
   C = (F-32)/1.8
4. Write a program to determine sum of the following series for given value of n:
   (1 + 1/2 + 1/3 + …… + 1/n). Print the result up to two decimal places.
5. [*] Write a program to find the *sum of digits* and *reverse* of a given integer number (take input using command-line argument).
6. Write a program to find the factorial of a given integer number using recursion (take input using command-line argument).
7. Write a program to show Fibonacci series up to n-th terms using recursion.
8. Write a Program of Sum of Series ($1+x+x^2+x^3+x^4+$…… up to n-th terms).
9. Write a program to calculate the simple interest (si) while your inputs are principle (p), time in years (n) and rate of interest (r) [take input using command-line argument].
10. [*] Write a program to find the real roots of the quadratic equation $ax^2 + bx + c = 0$ where a, b and c are constants.
11. WAP to print all prime number within a given range.
12. WAP to calculate GCD of two numbers.
13. Write a program to show addition, subtraction and multiplication of two matrices.
14. Write a program to find sum and average of several integers (in an array) using enhanced-for loop.

Assignment-2 (one day)

Objective:
*The objective of this assignment is to learn about **Classes** and **Objects** concept.*

1. Add two numbers by taking input using Command Line Input, Scanner class and BufferedReader class.
2. Write a program to find surface area and volume of Cylinder Using Constructors - Keyboard Input or Command Line Input.
3. Write a program to find surface area and volume of Cone Using Constructors - keyboard input or command line input.
4. Create a class First, make instance variable [int x], method [void show()] and also put main method inside that class and use the instance variable and method from main.
5. Create a class; make its instance variable and method. Use them from main, declared in different class.
6. Create a class named Test, make method inside it and pass object as parameter of this method using (a) pass same class's object, (b) pass different class's object.
7. Create a class; put a method inside this class which will return a class reference return same class and/or different class object.
8. [*] See the problems below:

   a) Write a JAVA Program to make a Student class with proper attributes like roll no, name, stream, and college. From main create such two students and show their information.

   b) Write a JAVA Program to consider the Student class in the previous Program. Assume that a student studies 6 subjects. Each subject has a title, internal marks and theory marks. Write a Program to define Student class including the subjects as array. From main create such two students and show their information including subjects' name and grand total marks.

   c) Write a JAVA Program to consider the Student class in the first Program 8a. Assume that students study varying number of subjects. Each subject has a title, internal marks and theory marks. Write a Program to define Student class including the subjects as ***vararg*** argument of constructor. From main create such two students and show their information including subjects' name and grand total marks.

9. [*] Design a class to represent a *Bank Account*. Include the following things:

   Fields

   - Name of the depositor
   - Address of the depositor
   - Account number
   - Balance amount in the account

   Methods

   - To assign initial values
   - To deposit an amount
   - To withdraw an amount after checking balance
   - To display the name, address and balance of a customer

   From main create object and call these methods.

Assignment-3 (two days)

Objective:

The objective of this assignment is to learn about ***inheritance, polymorphism, and abstract*** *classes.*

1. Create a class and determine if method overloading holds good for return type of methods or not.
2. Overload the constructors for classes Area and Volume of a rectangular figure and also display its area and volume. Area is the superclass and Volume is the subclass.
3. [*] Create a class **Employee** is having instance variables *name* and *id*. Create its subclass named **Scientist** which has instance variables *no_of_publication* and *experience*. Now create its subclass, say **DScientist** which has instance variable award. Put a method like: `public String toString(){ }` in every class where you describe about the class and from main() method create object of each class and print each object.
4. Create a class with a method `void show()` and make three subclasses of it and all subclasses have this show() method overridden and call those methods using their corresponding object references.
5. [*] Do the problem 4 using dynamic method dispatching.
6. Check without having any abstract method/s whether a class can be abstract; if so, then use that concrete method(s) from another class having main method.
7. Create an abstract class with three abstract methods check whether you can we override only few methods (not all methods) in subclass or not.
8. Assume that a bank maintains two kinds of account for its customers, one called savings account and other called current account. The savings account provides compound interest and withdrawal facilities but no cheque book facility. The current account provides cheque book facility but no interest. Current account holders should also maintain a minimum balance (say Rs. 1000) and if the balance falls below this level a service charge is imposed (say Rs. 100).

   Create a class **Account** that stores customer name, account number and type of account. From this class derive two classes **Curr_Acct** and **Savn_Acct** respectively to make them more specific to their requirements. Include the necessary methods to achieve the following tasks:

   a) Accept deposit from a customer and update the balance.
   b) Display the balance.
   c) Compute and deposit interest.
   d) Permit withdrawal and update the balance.
   e) Check for minimum balance, impose penalty, if necessary, and update balance.

   Use constructors to initialize the class members.

9. [*] Create a class Parent having instance variables id, name and address. Create a class ChildOne having instance variables id, name, address and marks. Also create another class ChildTwo with instance variables id, name, address, qualification and salary.Within each class define your own method to display values of these variables. Design the program using super call with proper parameter and use object of each class from main() to display their properties.

Assignment-4 (two days)

Objective:
*The objective of this assignment is to learn i**nterface, inner class, blocks and package** concept.*

1. Create an interface named Shape with a field pie (=3.14). Create two subclasses of it named Circle and Rectangle create object of the two classes and calculate their area.
2. Create a class which contains an inner class. Show that inner class can use member of outer class directly, but Outer class can use member of Inner class only through its object. Check the name of class file, you created.
3. Create two interfaces, each with two methods. Inherit a new interface from the two, adding a new method. Create a class by implementing the new interface and also inheriting from a concrete class. In main() method, create an object of derived class and call the methods [do all without package statement].
4. Write a program to demonstrate anonymous inner class (using super class and interface).
5. Show that ordinary block is executed when object is created and also the order of execution of these blocks (for multiple blocks/ inherited block).
6. Show that static block is executed at the time of class loading and also the order of execution of these blocks (for multiple blocks/ inherited block).
7. Write a program to show the difference between ordinary block and static block.
8. Write a program to demonstrate the order of execution among the parent and child's static and non-static blocks.
9. [*] Create a class with variable(s) and method(s) (all will be default accessed) under package pOne. Now create a class under package pTwo, which is subclass of firstly created class. In the method here (i.e. class of pTwo) call variable(s) and method(s) of previous class (i.e. class of pOne). If errors come, rectify them. Now from Main (under working directory) access second class's members.
10. Create an interface containing three methods, in a package 'pkgOne'. Implement the interface from a class under package pkgTwo. From main, under working directory, create object of the class and call methods of interface.

Assignment-5 (two days)

Objective:
*The objective of this assignment is to learn **String, Collection and File** concept.*

1. Take a sting from keyboard and convert into character array (new one).

2. Take a string from keyboard and a char array (filled up to length 5). Now append the string to that char array. Show the char array.

3. Find length of a string taken from keyboard and also find the length of that string except front and end spaces.

4. Check if "Tech" presents in "University of Technology" or not. If yes return its position.

5. Write a program to take a sentence and convert it into string arrays and sort the words using any sorting technique.

6. [*] Generate password from initials of one's first_name, middle_name, last_name and with last four digit of your roll_no (if middle name is not present, it won't come).

7. Write a program in Java which will read a string and rewrite it in the alphabetical order. For example, the word STRING should be written as GINRST.

8. Write a program in Java to extract a portion of a character string and print the extracted string. Assume that m characters are extracted, starting with the n-th character. The method signature will be like: **`void extract(String str, int n, int m).`**

9. Write your own method is having a signature like **`String deleteMe(String str, int m)`** that returns the input string with the m-th element removed.

10. Write a program to do the following:

i) To output the question "Who is the inventor of Java"?
ii) To accept an answer.
iii) To print out "Good" and then stop, if the answer is correct.
iv) To output the message "Try Again" and then stop, if the answer is wrong.
v) To display the correct answer when the answer is wrong even at the third attempts and stop.

11. Show that the String class type objects are immutable but StringBuffer class objects are mutable.

12. [*] Write a program in Java to create a StringBuffer object with content "Object Programming languages". Then perform the following operations:

i) Print the object and determine its length and capacity.
ii) Inset a new string "Oriented" after the word "Object". Print the modified string.
iii) Insert a character '-' at the 6-th position. Print the modified string again.
iv) Append another string named " are very popular." Print the resulting string.
v) Now delete the character '-' and put a space (" ") there. Print it.
vi) Apply these operations one-by-one: (a) `delete(1, 7)`, (b) `delete(2, 10)`, and (c) `delete(3, 13)`. Then print the result.
vii) Convert the StringBuffer type object into a String object. Print the final result.

13. Write a program in Java that checks whether a given string is a palindrome or not. Ignore the cases.

14. Write a program in Java that converts a string vector (containing strings) into an array of strings and display them [use command-line argument].

15. Write a program in Java that accepts a shopping list of five items from the command line and stores them in a vector.

16. Modify the program of Question No. 15 to accomplish the following:

- To delete an item in the list
- To add an item at a specified location in the list.
- To add an item at the end of the list.
- To print the contents of the vector.

17. Write a program to read words from a text input file and print them over console.

18. Write a program to copy the contents of one text file into another.

19. Write a program to save the sentences to a file you type using keyboard on a console until you put 'eof' marker (say '@').

20. Write a program to concatenate the contents of two files into a third one.

Assignment-6 (two days)

Objective:

*The objective of this assignment is to learn **Exception Handling** concept.*

1. Write a program to handle the ArithmeticException.

2. Write a program for multiple catch to fire ArrayIndexOutOfBoundsException and StringIndexOutOfBoundsException both.

3. Write a program to fire the NegativeArraySize exception.

4. Define an object reference and initialize it to null. Try to call a method through this reference. Now wrap the code in a try-catch clause to catch the exception.

5. [*]Write a program in Java to create a user defined exception named PayOutOfBoundsException (provided the monthly salary of a person is less than Rs. 10,000 /-) and fire the exception.

6. Write a program to fire any checked exception manually using 'throw' keyword.

7. [*] Create a class with two methods, f( ) and g( ). In g( ), throw an exception of a new type that you define. In f( ), call g( ), catch its exception and, in the catch clause, throw a different exception (of a second type that you define). Test your code in main( ) method.

8. Write a program that takes one string and two integers as command line argument and prints the reverse of the substring of the string specified by the two numbers. The program should handle all possible exception that may arise due to bad input.

9. Write a demo program to illustrate the restrictions of using 'throws' clause in method overriding with regard to superclass-subclass concept.

Assignment-7 (two days)

Objective:

*The objective of this Assignment is to learn **Multi-Threading** concept.*

1. Inherit a class from Thread and override the run( ) method. Inside run( ), print name of thread , and then call sleep( ). Repeat this three times, then return from run( ). Put a start-up message in the constructor. Make your thread object and main thread run to see what happens.
2. [*] Implement a class from Runnable and override the run( ) method. Inside run( ), print full qualified name of thread, and then call sleep( ). Repeat this three times, then return from run( ). Put a start-up message in the constructor. Make your thread object and main thread run to see what happens.
3. Make several threads (say 5) with names (by extending thread), set their priority and run them to see what happens.
4. Make several threads (say 5) with their names (implementing Runnable) set their priority and run them to see what happens.
5. Write a program to use join() and isAlive() in Multi-Threading.
6. Implement program of locking of common method by several threads. (Using the keyword 'synchronized').
7. Write a program to implement inter-thread communication: the consumer consumes items produced by the producer with proper synchronization.

Assignment-8 (two days)

Objective:

*The objective of this assignment is to learn **Applet** Concept.*

1. Display your name at middle portion of applet.
2. Draw an oval with centre at middle of applet and radius is 250 pixels.
3. Draw a circle of radius 200 pixels filled with blue color and center at the middle.
4. Generate a triangular wave on applet.
5. [*] Show phases of life cycle of applet using string message display both on console and applet.
6. Display table of 2 using font Tahoma with size 25 and bold letters on a black rectangular area with white font.
7. Programs of implementing PARAM concept using Applet.
8. [*] Display of banner using Applet.
9. Running Audio files (wav file) using Applet.
10. Implement a Calculator with Buttons using Applet.
11. Embedding of image files using Applet.
12. Display Sun beam from any corner of applet.

Assignment-9 (one day)

Objective:

*The objective of this Assignment is to learn **Swing** concept.*

1. [*] Write a program for simple calculator using swing.
2. Write a program to display system clock (digital) with time continuously updated per second.
3. Write a JAVA Program to create a Frame with Six Buttons representing your favorite six colors. When button is clicked, the background must be change to the corresponding colors.
4. Write a JAVA Program to create a window with four text fields for the Name, Street, City and Pincode with suitable labels. Also the window contains a button MyDetails. When the user types the Name, Street, City and Pincode and then clicks the button, the typed details must appear in Arial Font, size 30, Italics.
5. Write a JAVA Program to create a Frame with three text fields for Name, Age and Qualification and a text field of multiple lines for address.
6. Write a JAVA Program to create a window with a TextArea and two TextFields. The TextFields are called Find and Replace respectively. There is a button called Find and Replace. The user types a paragraph in the Text area, now types a word in the Text field Find and another text in the text field Replace. Now the users click the button. On pressing the button the paragraph in the text area is subject to the Find and Replace activity.
7. Write a JAVA Program to create a List Box with come colors in the items. When the color is selected, the background color must change accordingly. When the background color changes, the name of the color must also be displayed in the screen in Font Arial, 32, Bold + Italics.