**PSG COLLEGE OF TECHNOLOGY, COIMBATORE**

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**Ex7: INHERITANCE**

Define appropriate constructors, call to the superclass constructors, get and set methods for all the given problems.

1. Create a class, "PrimeTester", that defines one method "Boolean isPrime(int n)".When this function is called the implementing class should determine if the given "n" is prime or not. Create a class "SimpleTester" that inherits "PrimeTester" and uses a naive method to determine if a given number is a prime or not.
2. Create a class called Fruit with a method Boolean hasAPeel(). Write another class Vegetable with the method Boolean isARoot(). Create a class called Checking which consists of the following methods:

String doesThisHaveAPeel(Fruit f) and String  doesThisHaveARoot(Vegetable v).

The doesThisHaveAPeel() should in turn check hasAPeel(). If true display “This has a peel”. Else display “This doesn’t have a peel”. Similarly the doesThisHaveARoot() should check the isARoot(). If yes, display a message “This sis a root” else display “This is not a root”.

Create a class called Tomato which implements the above classes and provides proper implementation for the above methods.

1. Create a class called **ITEMS** which should have all the item names with its code. Create another class called **DETAILS** which contains the methods getPrice(int code) and getName(int code) to return the price and name of a particular item code. Create a class **SHOP** that inherits the classes **ITEMS** and **DETAILS**. Define the main method that should ask the user to enter the item code and display the product name and price based on the item code.
2. Create a class “Message” consisting of a private string data member “information”. Write set and get functions for assigning and returning value of the attribute “information”. Define a class called “Crypto” consisting of a member functions “Vignere\_cipher” to generate the cipher text. The vignere\_cipher technique uses multiple character keys .Each of the keys encrypts one single character. Each character is replaced by a number (A=0, B=1, …Z=25). After all keys are used, they are recycled. For encryption, Formula used : E=(M+K)mod 26

Plaintext: ATTACKATDAWN

Key: LEMONLEMONLE

Ciphertext: LXFOPVEFRNHR

Define class “Display” to print the cipher text and the plain text (original information).

1. Define a class “Book” to store the details author, book name, price, status, rack number and edition. Create display book function to display the details of the book and an update function to update the details of book. Create a class called “Librarian” having members name and password. Define a function search book, return book and issue book.
2. Create a class “Tree” consisting of member functions to create and display a binary search tree. Write a class “Difference” containing a function absolute to find the minimum absolute difference between the given two binary search trees. Define a class “Weight” which assign a weight to each node in the BST. Write a function exclusive\_or to find the node whose node value xor with weight gives the minimum value.

Minimum Absolute difference:

Example:

Input : N1 = 7, N2 = 2

BST1 :

5

/ \

3 7

/ \ / \

2 4 6 8

BST2 :

11

\

13

Output : 3

8 is largest number in the first BST

and 11 is smallest in the second.

Thus, the final answer will be 11-8 = 3

Input : N1 = 4, N2 = 2

BST1 :

3

/ \

2 4

\

14

BST2 :

7

\

13

Output : 1

1. Consider a superclass PurchaseItem which models customer’s purchases. This class has:

- two variables name (String) and unitPrice (double).

- One constructor to initialize the instance variables.

- A default constructor to initialize name to “no item”, and unitPrice to 0

- A method getPrice that returns the unitPrice.

Consider two subclasses WeighedItem and CountedItem. WeighedItem has an additional instance variable weight (double) in Kg while CountedItem has an additional variable quantity (int).

- Write an appropriate constructor for each of the classes making use of the constructor of the superclass to initialize its members.

- getPrice method that returns the price of the purchasedItem based on its unit price and weight (WeighedItem), or quantity (CountedItem). Make use of getPrice of the superclass.

Write an application class where you construct objects from the two subclasses and print them on the screen.

1. Consider a class “Dictionary” to represent a dictionary of words defined as a private data member and a class “Longest\_String” which has a string data member. Write appropriate functions to receive the input string and a function “Largest\_word” to find the largest word in the dictionary by deleting some characters of the given string.

Input : dict = {"ale", "apple", "monkey", "plea"}

str = "abpcplea"

Output : apple

Input : dict = {"pintu", "geeksfor", "geeksgeeks",

" forgeek"}

str = "geeksforgeeks"

Output : geeksgeeks

1. Define a class “Input” containing a set member function to receive input of a integer array. Write a class “sub-array” consisting of Maximum\_subarray function. This function given an array arr[] of N integers and an integer K it finds the maximum sub-array sum by flipping signs of at most K array elements.

Input: arr[] = {-6, 2, -1, -1000, 2}, k = 2

Output: 1009

We can flip the signs of -6 and -1000, to get maximum subarray sum as 1009

Input: arr[] = {-1, -2, -100, -10}, k = 1

Output: 100

We can only flip the sign of -100 to get 100

Input: {1, 2, 100, 10}, k = 1

Output: 113

We do not need to flip any elements