Subject : OOP with C++

Lab No : 1

Objective: Understanding the fundamental concept of class and object, static data members and static function members.

Key Terms: Class, Objects, Access specifiers, Data Hiding, Encapsulation, Static Data members, Static function members.

1. WAP to calculate the area of a rectangle. Create a class *Rectangle* having its private data members *length* and *breadth* and public member function: *getdata()* that receives data and *calculateArea()* that display the area.

   // Also, try to access the private members directly by the objects and observe the consequences.

2. WAP to calculate the area of a rectangle. Create a class *Rectangle* having its private data members *length* and *breadth* and public member function: *getdata()* that receives data and *calculateArea()* that display the area. The function members should be defined outside the class.

3. WAP to swap two numbers. Create a class *Demo* having its private data members *num1* and *num2* and public function member *swapdata(int, int)* that displays the swapped data. The *main()* function takes the two data and passes to *swapdata()* as arguments.

4. Create a class to represent batsman in a cricket team which includes the following members:

   Data member: Name, Runs made, No-of-fours, No-of-sixes

   Member function: To assign the initial value

   To display batsman information

   Create an array of object to display the information of five batsmen.

5. WAP to enter a number and print it in reverse order. Create a class *Calculate* with necessary private data members and two function members getnum() to accept an integer from user and *int reverse()* that returns the reverse number to the calling function.

6. WAP to define a class *Box* which has private data members *length*, *breadth* and *height*. And public function members: *readData()* for reading data members and *showVolume()* to calculate and return volume of box. The Volume should be displayed in the *main()*.

7. Create a class Demo with a non-static data member *num* and static data members *count*. The program should have a function members *getdata()* to take a number and display its square value. Also there should be a static function member *showcount()* to display for how many times the *getdata()* function have been called. Create as many objects as you desire.

8. Create a class *Person* with data member *name*, *age*, *address* and *votingID*. Write a function to initialize the *name, age and address* of person. Assign *voting ID* if the age of person is greater than 18. Otherwise, assign value 0 to *votingID*. Also create a function to display the values. The *votingID* starts from *1001*.

9. Create a class called employee which contains private data and member function like:

   Employee-Name, Employee- address, Employee-Phone, Salary, Net-salary, tax and calculate(), to calculate the Net-salary i.e.(Net-salary=Salary-tax) and public member functions like input() and output() to get the information of employee and display the information along with the calculated Net-salary value respectively. Now, write a main() program to create an object to handle 1 employee.

10. Write a class item with data item number and a static member variable count. Define a member function getdata() to accept input from user as well as count the number of times this function is called by different objects. Write a main function to exercise this class.

Subject: OOP in C++

Lab 3: Constructor and Destructor

... types of constructors, destructor and message passing

... constructor, destructor

Subject : OOP with C++

Lab No : 2

Objective : Understanding about friend function, friend class, default *arguments* and *reference* variable.

Key Terms : Friend function, Friend class, Default arguments, Inline function, Reference variable

1. WAP that has a class *Compare* with *n1* and *n2* as private data members, *getvalue()* as public member to accept two numbers from user. Make a friend function *compare()* to compare those two numbers and display the greatest among them.
2. There are two classes *GCES* and *PEC*. Each has a private data member (ie.no of students). Create a function in each class to take the value of number of students. Make a friend function to compare who has maximum no. of students.
3. Create a class called **time** with private data items hours, minutes and seconds. Write inside the class an input function that accepts inputs from the user. Write another outside the class definition that takes as arguments two objects of class **time** and returns another **time** object that holds the sum of the two **time** variables passed as arguments.
4. Create two classes DM and DB which store the value of distance. DM store distances in meters and centimeters and DB in feet and inches. Write a program that can read values for the class objects and add one object of DM with another object of DB.
5. There are two mutual friend classes *PU* and *TU*. PU has a private data member (ie. No of Phd students) and TU has a private data member (ie. No of researchers). Also PU has a function *showtotal()* which displays the total no of students in both PU and TU. Also TU has a function *showdifferences()* to display the difference of students in PU and TU. Write the program with necessary assumptions.
6. Write a program to create two classes: *PrithviHighway* and *MahendraHighway* both having with two private data members *km* and *mtr*. Each class have individual function members to take the length of highways in kilometer and meter. There is a friend function *showtotal()* to add the length of both highways. This friend function should be an inline function. Write a C++ program.
7. The default rate for simple interest for Fixed Deposit by Nepal Rastra Bank is 10 percent per annum. However, the commercial banks change their rate of interest depending upon the time period and principal amount. Now, write a C++ program using OOP concept, to receive principal amount and time and rate from user. Keeping default argumented rate, calculate the simple interest for n customers.
8. Write a program of your own to show the concept of reference variable.
9. Define a class to represent a bank account. Include the following members:

Data members: Name of the depositor, Account Number, Type of account, Balance amount in the account

Member Function: To assign initial values, To deposit an amount, To withdraw an amount after checking the balance, To display name and balance

Write a main program to test the program.

Subject: OOP in C++

Lab 3: Constructor and Destructor

Objective: Implementation of different types of constructors, destructor and message passing

Key Terms: Default constructor, copy constructor, parameterized constructor, destructor

1. Write a program to print the names of students by creating a Student class. If no name is passed while creating an object of the Student class, then the name should be "Unknown", otherwise the name should be equal to the String value passed while creating the object of the Student class.
2. Create a class named 'Rectangle' with two data members- length and breadth and a function to calculate the area which is 'length*breadth'. The class has three constructors which are:
   a - having no parameter - values of both length and breadth are assigned zero.
   b - having two numbers as parameters - the two numbers are assigned as length and breadth respectively.
   c - having one number as parameter - both length and breadth are assigned that number.
   Now, create objects of the 'Rectangle' class having none, one and two parameters and print their areas.
3. Suppose you have a Bank account with an initial amount of $50 and you have to add some more amount to it. Create a class 'AddAmount' with a data member named 'amount' with an initial value of $50. Now make two constructors of this class as follows:
   a - without any parameter –to assign initial value
   b - having a parameter which is the amount that will be added to the account
   Create an object of the 'AddAmount' class and display the final amount in the account.
4. Create a class to print the area of a square and a rectangle. The class has two functions with the same name but different number of parameters. The function for printing the area of rectangle has two parameters which are its length and breadth respectively while the other function for printing the area of square has one parameter which is the side of the square
5. WAP to find the simple interest. (Pass default argument rate in constructor).
6. WAP that copies data member of one object to another with the help of a copy constructor.
7. WAP to calculate the sum of integer numbers using constructor overloading.
8. Create a class called time that has separate int data member for hours, minutes and seconds. One constructor should initialize this data to 0 and another should initialize it to fixed values. A member function should display it in hh:mm:ss format. The final member function should add two objects of time passed as arguments. Create two initialized time objects and one that isn't initialized in the main program. Then it should add the two initialized values together leaving the result in the third time object.
9. WAP to add two complex numbers using the concept of constructor.
10. Create a class called Mountain with data members name, height, location and a constructor that initializes the members to the values passed to it as parameters, a function called CmpHeight() to compare two objects and DisplayInfo() to display the information of Mountain. In main, create two objects of the class mountain and print the information of the mountain which is greatest height.

Subject: OOP with C++

Lab 4: Inheritance

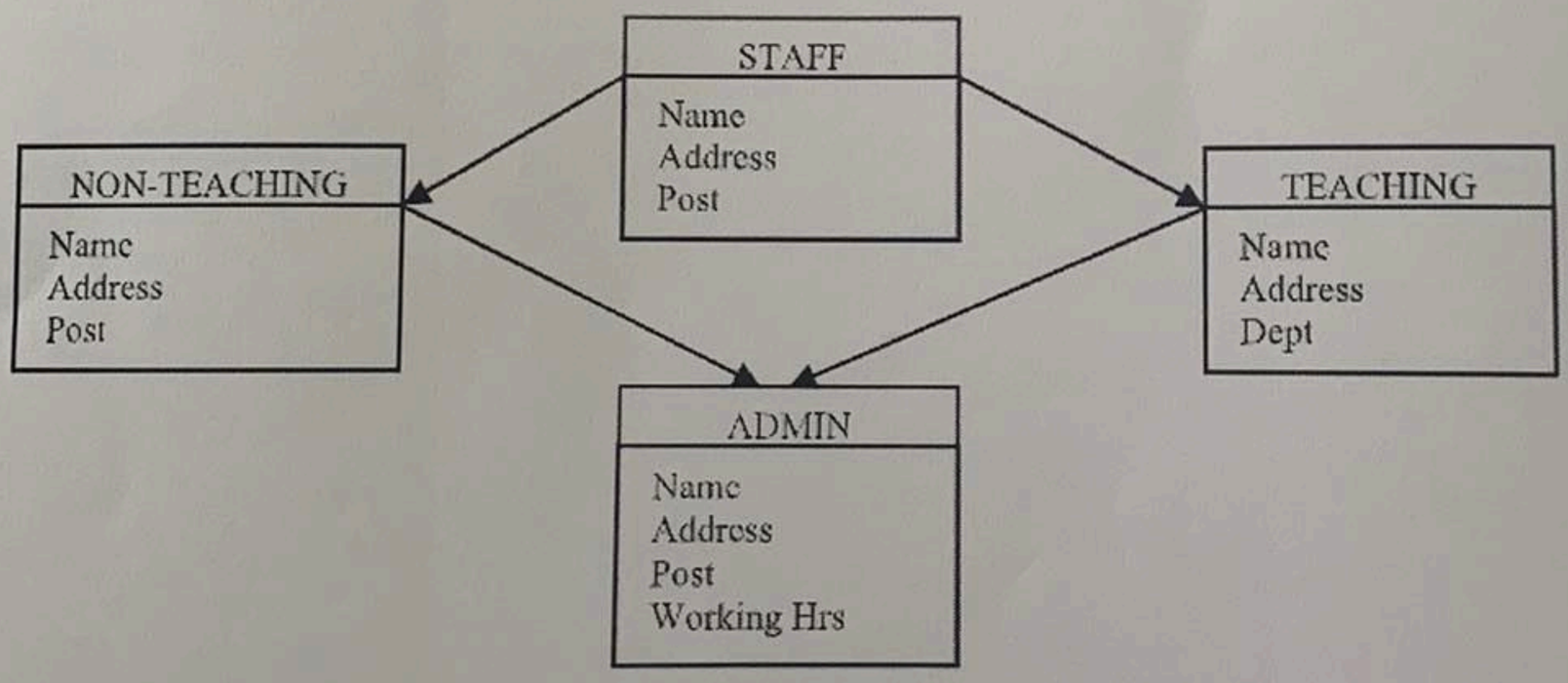
Objective: Understanding about different types of inheritance

Key Terms: inheritance, subclass, superclass, ambiguity, overriding, virtual function, pure virtual function

1. A class Room consists of two fields length and breadth and method int area() to find the area of room. A new class BedRoom is derived from class Room and consist of additional field height and two methods setData (int,int,int) to set the value for three fields and int volume() to find the volume. Now write the c++ program to input the length ,breadth and height and find the area and volume.
2. Create two classes class1 and class2 each having data member for storing a number, a method to initialize it. Create a new class class3 that is derived from both class class1 and class2 and consisting of a method that displays the sum of two numbers from class1 and class2.
3. A class Student consists of field roll, a method to assigns roll number. A new class Test is derived from class Student and consists of two new fields sub1 and sub2, a method to initialize these fields with obtained mark. Again, a new class Result is derived from Test and consists of a field total and a method to display entire details along with total obtained marks. WAP to input roll number, marks in two different subject and display total.
4. Create a class Student with data member roll_no and two functions to initialize and display it. Derive a new class Test which has two methods to assign and display marks in two subjects. Create a new class Sport with two functions that assign and display the score in sports. Now create another class Result that is derived from both class Test and Sport, having a function that displays the total of marks and score. Write a main program to test your class.
5. Define a shape class (with necessary constructors and member functions) in Object Oriented Programming (abstract necessary attributes and their types). Write a complete code in C++ programming language.
   i. Derive triangle and rectangle classes from shape class adding necessary attributes.
   ii. Use these classes in main function and display the area of triangle and rectangle.
6. WAP to illustrate a function overriding and pure virtual function. Make a class vehicle as base class. Car and Bus as derived classes from base class bus. Illustrate the concept of function overriding and pure virtual functions.
7. Create a class Student with data member roll_no and two functions to initialize and display Derive two new classes Theory and Practical from Student. Define suitable functions to assign and display theory and practical marks for two different subjects. Again, derive a new class

Result from both class Theory and Practical and add a new function to display the final total marks of student. Write a main program to test your class.

8. WAP to create class Student having data members name and roll number and necessary member functions. Again create another class Marks having data member's oop, pm, bc, acc,fin and also add necessary functions. Derive the class Result from student and marks having its own data member's total and percentage and member functions for calculating and displaying the data members. Finally create an object of Result class and then read and display its record.
9. Create class employee with data member's eid, ename, and salary and also add necessary member functions. Create another class company with company name, location and a employee. Add necessary member functions to the company class. Finally, create an object of company class and then read and display its record along with employee record.
10. The following figure shows minimum information required for each class. Write a program with member functions to read and display information of individual object. Every class should contain at least one constructor and should be inherited to other classes as well.



Gandaki College of Engineering and Science
Lamachaur, Pokhara

Object Oriented Programming in C++

Lab 6: Polymorphism & Template

1. Write a program showing "++" and "--" operator overloading (using member function and friend function).
2. Write a program to overload multiplication operator (*) showing the multiplication of two objects. (using member function and friend function).
3. Write a program to find the sum and difference of any two complex numbers by overloading '+' and '-' operator. (using member function and friend function).
4. Write a program to read a height of a person in Feet and Inches and convert it into Meter using user defined to class type conversion method. 1 meter =3.28084 feet, 1 feet = 12 inch.
5. Write a program to convert an object of polar class into the object of Rectangle class by using type conversion routine.
6. Create a template to find the sum of two integers and floats.
7. Create a template function to swap two values.
8. Write a function template to calculate the average and multiplication of numbers.