**LALITPUR ENGINEERING COLLEGE**

**DEPARTMENT OF COMPUTER ENGINEERING**

**SUBJECT: OBJECT ORIENTED PROGRAMMING IN C++**

**LABSHEET-5**

**Objective:** To familiarize students with polymorphism concepts.

**Polymorphism** is the ability to use an operator or function in different ways. It gives different meanings or functions to the operators or functions.

**Varieties of Polymorphism (Types)**

1. Compile time (Function overloading and operator overloading)

2. Run time (virtual function)

**Function** **Overloading**: Defining multiple functions with same name is known as function overloading. Arguments in the overloaded functions must not be same in number i.e. same function name different number of arguments.

**Operator Overloading:**

It is a mechanism that provides the operators a special meaning certain usage

**Syntax:**

return-type operator symbol (arguments)

{

Define Task of operator

}

**Type Conversion**

Compiler only supports basic to basic conversion such as int to float / int to char. Since object are user defined data type we must design conversion for that user defined data types. Three conversions are required in OOP

1. Basic to User defined (eg: int to Object)

2. User defined to Basic (eg: Object to int)

3. User defined type to another user defined type

**Programs:**

1. Write a program to generate Fibonacci series using operator overloading of ++ operator

a) for pre increment

b) for post increment

1. Write a program to implement vector addition using operator overloading
2. using friend function
3. without using friend function
4. Define two classes named polar and rectangle to represent points in polar and rectangle systems. Use conversion routines to convert from one system to another system.
5. Conversion routines in source class
6. Conversion routines in destination class
7. Write a program finding area of square, rectangle, triangle. Use function overloading technique.
8. Create a base class called shape. Use this class to store two double type values that could be used to compute the area of figures. Derive two specific classes called triangle and rectangle from the base shape. Add to the base class, a member function get\_data() to initialize base class data members and another member function display\_area() to compute and display the area of figures. Make display\_area() as virtual function and redefine this function in the derived class to suit their requirements.