Software Development Lab – II [15B17CI271]
Assignment Sheet
Week 4

COURSE OUTCOMES		COGNITIVE LEVELS
C173.1	Write programs in C++ to implement OOPs concepts related to objects, classes, constructor, destructor, and friend function.	Apply Level (Level 3)
C173.2	Write programs in C++ using OOPs concept like encapsulation, inheritance, polymorphism and abstraction.	Apply Level (Level 3)
C173.3	Write programs in C++ using Standard Template Library.	Apply Level (Level 3)
C173.4	Perform exception handling in C++ programs.	Apply Level (Level 3)
C173.5	Write MySQL queries to perform operations like ADD, DELETE, UPDATE, SELECT on relational databases.	Apply Level (Level 3)

Note: Students are advised to submit their solutions to respective lab faculty. The solution file must be named as "rollno_first name_w4.doc" (here w4 represents week4).

Q1. WAP in C++ to create a class Wall having private data members length and height. Create a parameterized constructor and a copy constructor to initialize these private data members. Define a member function to return the area. Demonstrate the working of each member function.

Q2. Create a class String with two private members (char * s; and int size;) to store a string and it's length. Define a constructor, a copy constructor and a destructor. Add a member function that prints the string. Demonstrate the working of each function.

Q3. What is the output of the following program?

```
#include <iostream>

using namespace std;

class Demo
{
        private:
                //static data members
                static int X;
                static int Y;

        public:
        //static member function
        static void  Print()
        {
                cout <<"Value of X: " << X << endl;
                cout <<"Value of Y: " << Y << endl;
        }
};

//static data members initializations
```

```
int Demo :: X =10;
int Demo :: Y =20;


int main()
{
        Demo OB;
        //accessing class name with object name
        cout<<"Printing through object name:"<<endl;
        OB.Print();

        //accessing class name with class name
        cout<<"Printing through class name:"<<endl;
        Demo::Print();

        return 0;
}
```

Q4. Define two classes ClassA and ClassB. ClassA has a private integer numA and ClassB has a private integer numB. Use friend function to add numA and numB of these classes.