Software Development Lab – II [15B17CI271]
Assignment Sheet
Week 5

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_w5.doc" (here w5 represents week).

Q1) Write a C++ program given that there are two base classes namely class A and class B from which class C is inherited. The class A contains member function getBase() and reads "Base" value as user input from keyboard. Class B contains member function getHeight() and reads "Height" value as user input from keyboard. The derived class C inherits all the public members of A and B and computes the area of the triangle.

SAMPLE OUTPUT:

enter value of base: 4.5

enter value of height: 78

area = 175.5

Q2) Write a C++ program, consider that there are two base classes namely class StudentsDetails and class Marks from which class C is inherited. The class A contains member function getDetails() that reads "students name", "Enrollment number" value as user input from keyboard. Class B contains member function getMarks() and reads "5 subject marks" value as user input from keyboard. The derived class C inherits all the public members of A and B and computes the area of the triangle.

SAMPLE OUTPUT:

enter value of name: JOHN

enter value of eno.: JOHN123

enter value of marks [0] 89

enter value of marks [1] 78

enter value of marks [2] 67

enter value of marks [3] 86

enter value of marks [4] 57

Total = 377

Q3) Based on the virtual function concept, write the main function for the following code to display the derived class values given by user at run time.

```
#include <iostream>
using namespace std;
class base {
  public:
   char fname[20];
   char surname[20];
public:
        virtual void calculate()
        {
        cout << "enter fname:";
        cin>> fname;
        cout << "enter surname";
        cin >> surname;
        }

        void display()
        {
        cout << "welcome" << fname << surname<<endl;
        }
};
class derived : public base {
public:
        void calculate()
        {
        cout << "enter derived_fname:";
        cin>>fname;
```

```
        cout << "enter derived_ surname";
        cin>>surname;
        }

        void display()
        {
        cout << "welcome to derived" << fname << surname<<endl;
        }
};
Int main()
{
WRITE YOUR CODE HERE.
}
```

Q4) Given a snippet of the program to create a base class named as *base_food_Items* with a virtual function named as *order* and *total_Price* . Create a derived class name *Chinese*. Then calculate the total_price of food items based on variables *quantity* and *item_price.*

```
#include <iostream>
using namespace std;
class base_food_items {
  public:
   char item_name[20];
   int quantity;
   int item_price;
public:
        virtual void order()
        {
        cout << "enter item name:";
        cin>> item_name;
        cout << "enter quantity";
        cin>> quantity;
```

```
        cout << "Item price";
                cin >> item_price;
        }
         void total_price()
        {
           cout<<"order is: " << item_name<<"\t"<<"quantity:"<<quantity<<endl;
                        cout << "total price=" << item_price*quantity<<endl;
        }
};
```

Q5) Write a C++ program to show the functionality of the abstract classes.

Output :

```
This is Display1() method of Derived Class
This is Display2() method of Derived Class
```

Q6) Write a program to use constructors of the abstract class to find the sum of two numbers and display the results.

Q7 A) What is the output of the following code?

```
#include<iostream>
using namespace std;
class Base { };
class Derived: public Base {};
int main() {
  Base *base_ptr = new Derived;
  Derived *derived_ptr = dynamic_cast<Derived*>(base_ptr);
  if(derived_ptr != NULL)
    cout<<"It is working";
  else
    cout<<"cannot cast Base* to Derived*";
  return 0;
```

```
}
```

Q7 B) Rewrite the above code to rectify the error