Everest Engineering College
Sanepa-2, Lalitpur

Date of distribution:…….
Date of submission:……..

Subject: Object Oriented Programming in C++

Lab-4

Title: Friend function and class

Objective: To be familiar with friend function and friend class

Theory:

Friend Function

The private member of a class cannot be accessed from outside the class i.e. a nonmember function cannot access to the private data of a class. But may be in some situation one class wants to access private data of second class and second wants to access private data of first class, or may be an outside function wants to access the private data of a class. However, we can achieve this by using friend functions.
To make an outside function friendly to a class, we have to declare this function as a friend Function Declaration of friend function should be preceded by the keyword friend.

Example:

```
#include<iostream>
using namespace std;
 class Sample
 {
     int a,b;
     public:
    void setvalue(int x,int y)
   {
     a=x;
     b=y;
   }
   friend void sum(Sample s);
 };

 void sum(Sample s)
 {
   cout<<"sum="<<(s.a+s.b)<<endl;

 }
```

```
int main()
{
  Sample s;
  s.setvalue(5,10);
  sum(s);
  return 0;
}
```

Friend class

We can declare all the member functions of one class as the friend functions of another class. In such cases, the class is called friend class. A friend class can use all the data member of a class for which it is friend. For example

```
class B;
class A
{
…….
friend class B;
};
```

Here, class B can use all the data member of class A

Example:

```
#include<iostream>
using namespace std;
class B;
class A
{
int x,y;
public:
void getdata()
{
cout<<"Enter the value of x and y:"<<endl;
cin>>x>>y;
}
friend class B;
};
```

```
class B
{
int z;
public:
void getdata()
{
cout<<"Enter the value of z:";
cin>>z;
}
void sum(A t)
{
cout<<"Sum of x,y and z ="<<(t.x+t.y+z)<<endl;
}
};
int main()
{
A p;
B q;
p.getdata();
q.getdata();
q.sum(p);
return 0;
}
```

Lab exercises (please code yourself and show the output to instructor)

1. WAP to create a single class with private data members and add them using friend function.
2. WAP to add private data of two different classes using friend function.
3. WAP to add private data of three different classes using friend function.
4. Write a program to find the largest of four integers. your program should have three classes and each classes have one integer number.
5. WAP to swap the contents of two variables of 2 different classes using friend function.
6. WAP to add complex numbers of two different classes using friend class.
7. Create classes called class1 and class2 with each of having one private member. Add member function to set a value(say setvalue()) one each class. Add one more function max() that is friendly to both classes. max() function should compare two private member of two classes and show maximum among them. Create one-one object of each class and then set a value on them. Display the maximum number among them.