Lecture NO 7

- **Use of Access Modifier: Protected**
- **How to define Function(s) outside the Class**

```
#include<iostream>
using namespace std;

class myClass{
	protected:
		int ProtectedVar = 0;
	private:
		float x=0,y=0;
	public:
		void setX(float pX);
		void setY(float pY);
		void getX();
		void getY();
		void setProtectedVar(float pY);
		void getProtectedVar();
		void Print();
};


		void myClass::setX(float pX)
		{
			if(pX < 0)
			{
				cout<<endl<<"Please enter a valid number";
			}
			else
			{
				x = pX;
			}
//			Print();	// we can call another function
		}

		void myClass::setY(float pY)
		{
			if(pY < 0 )
			{
				cout<<endl<<"Please enter a valid number";
			}
			else
			{
```

```
                        y = pY;
                }
        }

        void myClass::setProtectedVar(float pY)
        {
                if(pY < 0 )
                {
                        cout<<endl<<"Please enter a valid number";
                }
                else
                {
                        ProtectedVar = pY;
                }
        }

        void myClass::getX()
        {
                cout<<endl<<"X = "<<x;
//              cout<<endl<<"myPrrotectedVar = "<<myProtected;
        }

        void myClass::getY()
        {
                cout<<endl<<"Y = "<<y;
        }

        void myClass::getProtectedVar()
        {
                cout<<endl<<"ProtectedVar = "<<ProtectedVar;
        }

        void myClass::Print()
        {
                cout<<endl<<"It is my test class";
        }

int main()
{
        myClass myC1;

// we can't access a protected data memeber of a class from main directly
// unless we made the set() get() functions for that particular data member

//  myC1.ProtectedVar = 90;

//      myC1.getX();
//      myC1.getY();
```

```
        myC1.setX(12.2);
        myC1.setY(12.3);

        myC1.Print();
        myC1.getProtectedVar();
        myC1.getX();
        myC1.getY();
}
```