## ***Objectives***

After completing this lab, you will be able to:

1. Implement a base class and derived classes in C++
2. Understand the working of base class and derived class constructors/destructors.
3. Understand the working and purpose of private and protected attributes of a base class.
4. Use base class member functions in a derived class.
5. All the variables will be dynamic.

### ***Inheritance & Polymorphism***

### **Important Note:** For this lab write complete code in one cpp file. You do not need to make separate header and cpp files for each class.

### **Exercise 1:**

Consider the following hierarchy as it exists in a university:

* There is only one type of person in the university i.e. Student.
* Every Person has some basic information that is common to all persons i.e. the name and ID stored as protected attributes and age which is a protected attribute.
* A student can in turn be either an Undergraduate or a Graduate student, every student has a cgpa and rollNumber.
* An undergraduate student has a fyp\_name as his private attribute.
* A graduate student has a thesis topic as his private attribute.

All the data members are protected except from undergraduate and graduate class.

* Add appropriate constructors and destructors to all the classes. For example, the constructor for the Person class should take three inputs (*ID*, first\_name, last\_name and age).

### Add a member virtual function, virtual **void print()** in the Person class.

### Add a member virtual function, virtual **void print()** in the Student class. This method should print the name, cgpa and age of the student.

**Sample output:** “Ted Thompson is 22 years old, his cgpa is 3.91”

### Add a member function **void print()** in the Graduate class. This method should print the name, cgpa and age of the student and his thesis topic.

**Sample output for void print()**

“Ted Thompson is a graduate student, his cgpa is 3.91 and his thesis topic is Distributed Algorithms”

### Add a member function **void print()** in the UnderGraduate class. This method should print the name, cgpa and age of the student and his fyp\_name.

**Sample output for void print()**

“Ted Thompson is an undergraduate student, his cgpa is 3.91 and his final year project is titled The Even Locator”

### Add a member virtual function, virtual **void input()** in the Student class. This method should take input in ID, name, cgpa rollNumber and age of the student.

### Add a member function **void input()** in the Graduate class. This method should take input in thesis topic.

### Add a member function **void input()** in the UnderGraduate class. This method should take input in fyp\_name.

//example main function

void main()

{

//format (ID, fName, lName, Age, RollNumber, cgpa, fyp/thesis)

Student \* s = new Undergraduate(1,"Ted”, T”hompson",22, “14L-4171”, 3.91,"The Event Locator");

Student \* s2 = new Graduate(2,"Arnold”, “Gates",25, “17L-6171” 3.01,"Distributed Algorithms");

s->print();

s->input();

s2->print();

s2->print();

s->print();

s2->print();

}

Implement this main function:

1. Create two objects UnderGraduate and Graduate class individually.
2. Call the input function to take input in the data from user.
3. Create two pointers of class Student which will hold two objects of UnderGraduate and Graduate class. Call Input and Print functions.
4. Create a pointer of Person class which will hold an object from UnderGraduate/Graduate class (upon your wish). Call Print Function.