**Heaven’s Light is Our Guide**



**Rajshahi University of Engineering and Technology**

**Department of Computer Science and Engineering**

**Course No:** CSE.1204

**Course Title:** Sessional based on CSE.1203 (Object Oriented Programming)

**Lab Report No:** 07

**Lab Report On:** Inheritance.

**Submitted By** **Submitted To**

Md. Ariful Islam Md. Asifur Rahman

Roll No: 1803046 Lecturer

Section: A Dept. of CSE,RUET

Department: CSE

**Problem No:** 01

**Problem Statement:** Implementation of **Inheritance** in the following classes.

**class** student

|  |
| --- |
| private:  string roll;  float cgpa; |
| void print\_marks ( ); |

**class** person

|  |
| --- |
| private:  string name;  int age;  protected:  string mobile\_no; |
| void print\_info ( ); |

public

**class** teacher

|  |
| --- |
| private:  string teacher\_id;  string joining\_date;  string designation; |
| void print\_info ( ); |

private

**Theory**

The capability of a class to derive properties and characteristics from another class is called **Inheritance**. Inheritance is one of the most important feature of Object Oriented Programming. In inheritance there are two types of classes:

1. **Sub Class:** The class that inherits properties from another class is called Sub class or Derived Class.
2. **Super Class:** The class whose properties are inherited by sub class is called Base Class or Super class.

The syntax for creating a sub-class inherited from a base-class is given below:

**Class subclass\_name : access\_mode base\_class\_name**

**{**

**//body of subclass**

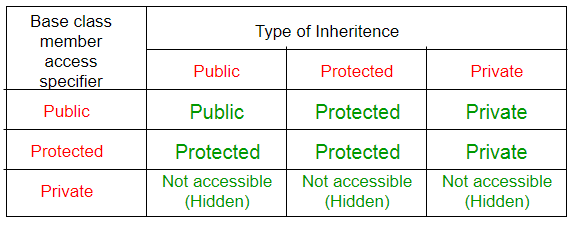
**};**

**Modes of Inheritance**

1. **Public mode**: If we derive a sub class from a public base class. Then the public member of the base class will become public in the derived class and protected members of the base class will become protected in derived class.
2. **Protected mode**: If we derive a sub class from a Protected base class. Then both public member and protected members of the base class will become protected in derived class.
3. **Private mode**: If we derive a sub class from a Private base class. Then both public member and protected members of the base class will become Private in derived class.

There is a private members in the base class cannot be directly accessed in the derived class, while protected members can be directly accessed.

The below table summarizes the above three modes and shows the access specifier of the members of base class in the sub class when derived in public, protected and private modes:



**Source Code**

1. **main.cpp :**

|  |
| --- |
| #include <iostream>  #include<string.h>  #include "person.h"  using namespace std;  int main()  {  person P1("Ashraful",21,"01553454534");  P1.print\_info();  teacher T1("NI",27,"0134345345","2211","20-01-2016","Lecturer");  T1.print\_info();  student S1("Sudipto",21,"01710158323","1803045",3.5);  S1.print\_info();  person \*p;  p=&S1;  p->print\_info();  return 0;  } |

1. **.h file :**

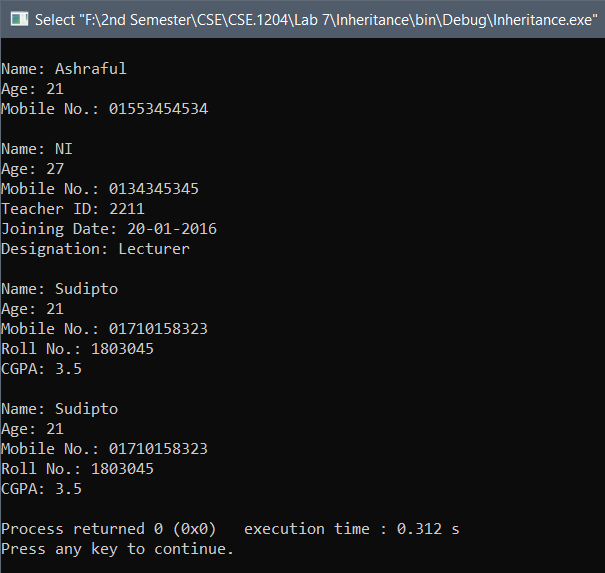
**person.h**

|  |
| --- |
| #ifndef PERSON\_H  #define PERSON\_H  class student;  class teacher;  class person  {  private:  char name[25];  int age;  protected:  char mobile\_no[25];  public:  person(char a[],int b,char c[]);  virtual void print\_info();  };  class teacher:private person{  private:  char teacher\_id[25];  char joining\_date[25];  char designation[25];  public:  teacher(char a[],int b,char c[],char d[],char e[],char f[]);  void print\_info();  };  class student:public person{  private:  char roll[25];  float cgpa;  public:  student(char a[],int b,char c[],char d[],float e);  void print\_info();  };  #endif // PERSON\_H |

1. **.cpp file**

**person.cpp**

|  |
| --- |
| #include <iostream>  #include<string.h>  #include "person.h"  using namespace std;  person::person(char a[],int b,char c[]){  strcpy(name,a);  age=b;  strcpy(mobile\_no,c);  }  void person::print\_info(){  cout<<endl;  cout<<"Name: "<<name<<endl;  cout<<"Age: "<<age<<endl;  cout<<"Mobile No.: "<<mobile\_no<<endl;  }  teacher::teacher(char a[],int b,char c[],char d[],char e[],char f[]):person(a,b,c){  strcpy(teacher\_id,d);  strcpy(joining\_date,e);  strcpy(designation,f);  }  void teacher::print\_info(){  person::print\_info();  cout<<"Teacher ID: "<<teacher\_id<<endl;  cout<<"Joining Date: "<<joining\_date<<endl;  cout<<"Designation: "<<designation<<endl;  }  student::student(char a[],int b,char c[],char d[],float e):person(a,b,c){  strcpy(roll,d);  cgpa=e;  }  void student::print\_info(){  person::print\_info();  cout<<"Roll No.: "<<roll<<endl;  cout<<"CGPA: "<<cgpa<<endl;  } |

**Output**

**Conclusion :** By our Course Teachers help and my knowledge about C and C++, I completed the program.

**# The End #**