

Department of Electronics & Telecommunication Engineering

Name of the Experiment:

Inheritance in C++.

Course No. : CSE-284
Course Title : Object Oriented Programming Sessional
Experiment No. : 04
Date of Exp. : 30-10-2024
Date of Submission : 13-11-2024

Remarks :

| |
|--|
| |
|--|

Name : Tasnia Afrin
ID : 2108050
Level : 02
Term : 02
Group : 02

Experiment Name :

Inheritance in C++.

Objectives:

- Familiarize with Inheritance.
- Explain the concept of single and Multi level inheritance in OOP.
- Solve various problems in order to comprehend the above topics.

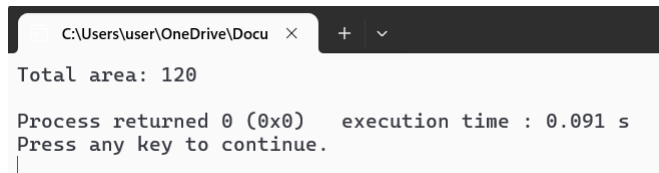
Example-1 :

A C++ program to demonstrate the single level inheritance.

Input:

```
#include<iostream>
using namespace std;
class Shape{
    protected:
    int width, height;
    public:
        void setWidth(int w){
            width = w;
        }
        void setHeight(int h){
            height = h;
        }
};
class Rectangle: public Shape{
public:
    int getArea(){
        return (width*height);
    }
};
int main(void){
    Rectangle rect;
    rect.setWidth(10);
    rect.setHeight(12);
    cout<<"Total area: "<<rect.getArea()<<endl;
    return 0;
}
```

Output :



```
C:\Users\user\OneDrive\Docu  ×  +  ▾
Total area: 120
Process returned 0 (0x0)   execution time : 0.091 s
Press any key to continue.
|
```

Example-2 :

A C++ program to demonstrate the Multilevel Inheritance.

Input:

```
#include<iostream>
using namespace std;
class base
{
public:
    int x;
    void getData(){
        cout<<"Enter value of x= "; cin>>x;
    }
};
class derive1: public base
{
public:
    int y;
    void readdata(){
        cout<<"\nEnter value of y= ";cin>>y;
    }
};
class derive2: public derive1
{
private:
    int z;
public:
    void indata(){
        cout<<"\nEnter value of z= ";cin>>z;
    }
    void product(){
        cout<<"\nProduct= "<<x*y*z;
    }
};
int main(){
    derive2 a;
    a.getData();
    a.readdata();
    a.indata();
    a.product();
    return 0;
}
```

Output :

```
C:\Users\user\OneDrive\Docu × + ▾
Enter value of x= 12
Enter value of y= 15
Enter value of z= 8
Product= 1440
Process returned 0 (0x0) execution time : 10.477 s
Press any key to continue.
```

Practice Exercise: 1

Write a C++ program to add two numbers. Accept these two numbers from the user in base class and display the sum of these two numbers in derived class.

Input:

```
#include <iostream>
using namespace std;
// Base class
class Base {
protected:
    int num1, num2;
public:
    void getNumbers() {
        cout << "Enter number1: ";
        cin >> num1;
        cout << "Enter number2: ";
        cin >> num2;
    }
};
// Derived class
class Derived : public Base {
public:
    void displaySum() {
        int sum = num1 + num2;
        cout << "The sum of the two numbers is: " << sum << endl;
    }
};
int main() {
    Derived obj;
    obj.getNumbers(); // Calling function from base class
    obj.displaySum(); // Calling function from derived class
    return 0;
}
```

Output :

```
C:\Users\user\OneDrive\Docu × + ▾
Enter number1: 11
Enter number2: 22
The sum of the two numbers: 33
Process returned 0 (0x0) execution time : 42.744 s
Press any key to continue.
```

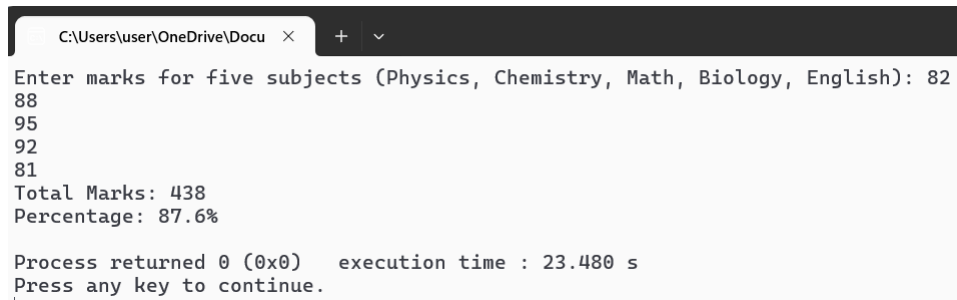
Practice Exercise: 2

Write a C++ program to calculate the percentage of a student. Accept the marks of five subjects (Physics, Chemistry, Math, Biology, and English) in base class. A class will derived from the base class which includes function to find the total marks obtained and another class derived from this first derived class which calculates and displays the percentage of student.

Input:

```
#include <iostream>
using namespace std;
// Base class
class Student {
protected:
    int marks[5];
public:
    void getMarks() {
        cout << "Enter marks for five subjects (Physics, Chemistry, Math, B
        for (int i = 0; i < 5; i++) {
            cin >> marks[i];
        }
    }
};
// Derived class to calculate total marks
class TotalMarks : public Student {
protected:
    int total;
public:
    void calcTotal() {
        total = 0;
        for (int i = 0; i < 5; i++) {
            total += marks[i];
        }
    }
    int getTotal() {
        return total;
    }
};
// Another derived class to calculate and display percentage
class Percentage : public TotalMarks {
public:
    void calcPercentage() {
        calcTotal();
        float percentage = (total / 5.0);
        cout << "Total Marks: " << total << endl;
        cout << "Percentage: " << percentage << "%" << endl;
    }
};
int main() {
    Percentage student;
    student.getMarks();
    student.calcPercentage();
    return 0;
}
```

Output :

A screenshot of a terminal window showing the execution of a C++ program. The window has a dark title bar with a tab labeled 'C:\Users\user\OneDrive\Docu' and standard window controls. The terminal text is as follows:

```
Enter marks for five subjects (Physics, Chemistry, Math, Biology, English): 82
88
95
92
81
Total Marks: 438
Percentage: 87.6%

Process returned 0 (0x0)   execution time : 23.480 s
Press any key to continue.
```

Discussion:

In this experiment above, inheritance in C++ were introduced. Here, single and multi level inheritance with base class and derived class were also familiarized. Different types of functions were created as requirement and their level of inheritance was observed. While solving the problems, I faced a little difficulty when deriving a class from another derived class. But after implementing part by part, the concept was understood, fixing the errors, the problems were solved as well.