B’ s default constructor called.
A’ s default constructor called.
A’ s default constructor called.
B’ s default constructor called.
A’ s constructor called.
A’ s constructor called.
B’ s constructor called.
B’ s destructor called.
A’ s destructor called.
A’ s destructor called.
A’ s constructor called.
A’ s constructor called.
B’ s constructor called.
B’ s destructor called.
A’ s destructor called.
A’ s destructor called.

三.1
(1,2)
5,6
(6,9)

三.2
(1,2)
(6,9)
5,6
(6,9)

1,5,2

3,7,4

B’ s destructor called.
A’ s destructor called.
A’ s destructor called.
B’ s destructor called.
A’ s destructor called.
A’ s destructor called.

2.实验心得：

通过实验，我深刻理解了 C++中继承性和派生类的应用。这个过程中，我不仅学会了如何使用继承性和派生类来重用和扩展代码，还学会了如何在不同的类之间实现多态性。这些技能对于提高代码的可维护性和可扩展性非常重要。

此外，我还发现继承性和派生类的使用可以帮助我更好地组织和管理代码。同时，通过派生类来扩展现有功能，我可以更容易地实现功能的定制和扩展。

综上所述，通过实验，我深刻认识到了继承性和派生类的应用对于提高代码质量和效率的重要性。我相信在未来的学习和工作中，我会更加频繁地使用这些技术来优化我的代码。

天津理工大学
Tianjin University of Technology

实验报告

课程名称：C++程序设计与算法实现
课程代码：0167473
实验名称：C++语言基础知识实验
姓　　名：吴睿奇
学　　号：20220397
班　　级：机器人一班
实验日期：2023.11.20

实验成绩记录表			
评价内容	优秀标准	分值	得分
实验预习	已完成对本次实验相关实验原理、实验设备使用情况的预习。已完成实验报告中实验名称、实验目的、实验设备、实验原理（或实验方案）、实验步骤等内容的填写。	15	
实验纪律及操作	在规定时间内提前完成实验操作，操作熟练，完全符合实验室安全规定。能够正确测量实验数据，并完整、规范地进行记录，记录纸撰写工整。能够主动将所有设备(包括导线等)复原、归位，具有良好的实验习惯。	15	
实验报告	实验方案合理、正确。实验报告撰写规范、内容完整无漏项。实验数据详实、准确、有效，能够合理分析实验结果，得到有效结论，并能提出改进。	70	
教师签字或盖章		总分	

机械工程学院

一、实验目的：

1、熟悉 C++语言的运行环境，程序的编辑、编译和运行的过程。

2、掌握继承中派生类的三种继承方式，单继承中成员访问权限的控制及派生类的构造函数和析构函数的使用。

二、实验设备：装有 VC6.0 运行环境的计算机。

三、实验要求：

1. 分析并验证讲义中有关“成员访问权限的控制”的应用程序（例 7.2），并回答所提的问题。

1） 执行该程序时，哪个语句会出现编译错 ？为什么 ？

程序中，d1.g();语法出现编译错，因为B是以私有继承方式继承类A的，所以B类的对象不可访问A类成员函数。

2） 去掉出错语句后，执行该程序后输出结果如何?

6

h

3） 程序中派生类 B 是从基类 A 继承来的，这种缺省继承方式是哪种继承方式 ？

使用class关键字定义类时，默认的继承方式是private。

4） 派生类 B 中，A::f 的含意是什么 ？

A::f;将基类中的公有成员说明为派生类的公有成员。

5） 将派生类 B 的继承改为公有继承方式该程序将输出什么结果 ？

6

g

h

2. 分析并验证讲义中有关“构造函数和析构函数”的应用程序（例 7.4）的运行结果。

3. 上机调试第七章作业题：“三．1”和“三．2”程序，分析运行结果。

四、实验步骤与数据

1. 实验步骤。

(1).创建工程

(2).创建源文件

(3).编写代码

2. 程序调试结果。

例 7.4:

```
#include <iostream.h>
class A {
public:
    A() {
        a = 0;
        cout << "A's default constructor called.\n";
    }
    A(int i) {
        a = i;
        cout << "A's constructor called.\n";
    }
    ~A() {
        cout << "A's destructor called.\n";
    }
    void Print() const {
        cout << a << ",";
    }
    int Geta() {
        return a;
    }
private:
    int a;  };
class B:public A
{
public:
    B()
    {b=0;cout<<"B s default constructor called.\n";}
    B(int i,int j,int k);
    ~B()
    { cout<<"B s destructor called.\n";}
    void Print();
private:
    int b;
    A aa;
};
B::B(int i,int j,int k):A(i),aa(j)
{
b=k;
cout<<"B s constructor called.\n";
}
void B::Print()
{
    A::Print();
    cout<<b<<","<<aa.Geta()<<endl;
}
void main()
{
B bb[2];
bb[0]=B(1,2,5);
bb[1]=B(3,4,7);
for(int i=0;i<2;i++)
bb[i].Print();
}
```

```
"C:\Users\B54\Downloads\Debug\12 (1).ex
B s default constructor called.
A's default constructor called.
A's default constructor called.
B s default constructor called.
A's constructor called.
A's constructor called.
B s constructor called.
B s destructor called.
A's destructor called.
A's destructor called.
A's constructor called.
A's constructor called.
B s constructor called.
B s destructor called.
A's destructor called.
A's destructor called.
1,5,2
3,7,4
B s destructor called.
A's destructor called.
A's destructor called.
B s destructor called.
A's destructor called.
A's destructor called.
Press any key to continue
```

三.1:

```
# include <iostream.h>
class A
{
public:
A(int i,int j){a=i; b=j;}
void Move(int x,int y) {a+=x;b+=y;}
void Show() { cout<<"("<<a<<","<<b<<")"<<endl;}
private:
    int a,b;
};
class B:private A
{
public:
B(int i,int j,int k,int l):A(i,j){x=k;y=l;}
void Show() { cout<<x<<","<<y<<endl; }
void fun( ) {Move(3,5);}
void f1() { A::Show();}
private:
    int x,y;
};
void main()
{
A e(1,2);
e.Show();
B d(3,4,5,6);
d. fun();
d. Show();
d.f1();
}
```

```
"C:\Use
(1,2)
5,6
(6,9)
Press an
```

三.2:

```
# include <iostream.h>
class A
{
public:
A(int i,int j){a=i;b=j;}
void Move(int x,int y){a+=x;b+=y;}
void Show( ) {cout<<"("<<a<<","<<b<<")"<<endl;}
private:
    int a,b;
};
class B:public A
{
public:
B(int i,int j,int k,int l):A(i,j),x(k),y(l)
{ }
void Show() {cout<<x<<","<<y<<endl;}
void fun() {Move(3,5);}
void f1() { A::Show();}
private:
    int x,y;
};
void main( )
{
A e(1,2);
e.Show( );
B d(3,4,5,6);
d.fun();
d.A::Show( );
d. B::Show();
d.f1();
}
```

```
"C:\Users\B
(1,2)
(6,9)
5,6
(6,9)
Press any k
```

五、实验结果与讨论

1、分析程序的理论运行结果，并与实际运行结果进行比较。

理论运行结果：

(1).例 7.4

A’s default constructor called.

A’s default constructor called.