

一、源程序

```
1.  #include <iostream>
2.  class Human
3.  {
4.  public://修改 1 处 依次修改为 private\protected
5.      int age;
6.  };
7.  class student:private Human////修改 2 处 依次修改为 private\protected
8.  {
9.  public:
10.     void showAge()
11.     {
12.         cout<<"the student age is"<<age<<endl;
13.     }
14. };
15. class Primary:public student////修改 3 处 依次修改为 private\protected
16. {
17. public :
18.     void setAge(int a)
19.     {age=a;}
20. };
21. int main()
22. {
23.     Primary jessic;
24.     jessic.age=12;
25.     jessic.showAge();
26.     return 0;
27. }
```

二、分别修改继承方式

2.1 验证 private、public 权限 与 公有、私有继承

2.1.1 基类公有成员，派生类公有继承

测试代码:

```
28.  #include <iostream>
29.  class Human
```

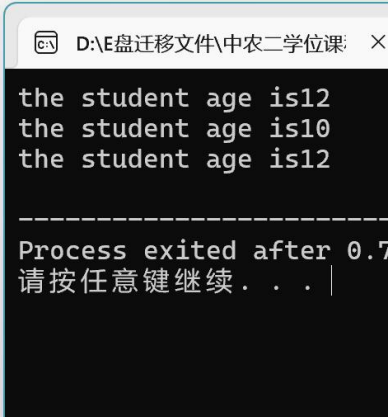
```

30.  {
31.  public://修改 1 处 依次修改为 private\protected
32.      int age;
33.  };
34.  class student:private Human////修改 2 处 依次修改为 private\protected
35.  {
36.  public:
37.      void showAge()
38.      {
39.          cout<<"the student age is"<<age<<endl;
40.      }
41.  };
42.  class Primary:public student////修改 3 处 依次修改为 private\protected
43.  {
44.  public :
45.      void setAge(int a)
46.      {age=a;}
47.  };
int main()
{
    Primary jessic;
    jessic.age=12;
    jessic.showAge();
    jessic.setAge(10);
    jessic.showAge();

    student jack;
    jack.age = 12;
    jack.showAge();

    return 0;
}

```



结论：基类的公有成员在派生类中可见且在派生类中的访问权限也为公有。

2.1.2 基类公有成员，派生类私有继承

测试代码：

```

48.  class Human
49.  {
50.  public://修改 1 处 依次修改为 private\protected
51.      int age;
52.  };
53.  class student:private Human////修改 2 处 依次修改为 private\protected
54.  {

```

```

55.     public:
56.         void showAge()
57.         {
58.             cout<<"the student age is"<<age<<endl;
59.         }
60.     };

```

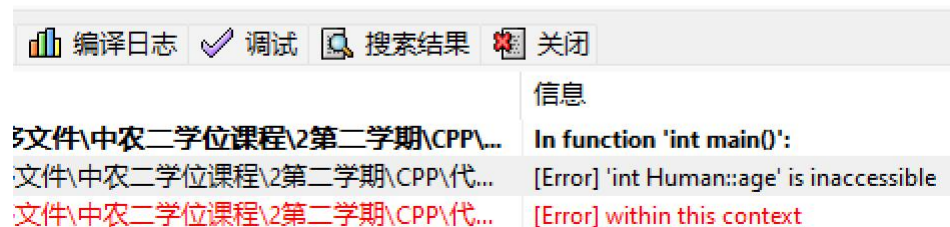
结果:

```

student jack;
jack.age = 12;
jack.showAge();

return 0;

```



结论：基类的公有成员在经过私有继承后，在派生类中可见，但在派生类中访问权限为私有。

2.1.3 基类私有成员

测试代码:

```

61.     class Human
62.     {
63.     private://修改 1 处 依次修改为 private\protected
64.         int age;
65.     };
66.     class student:public Human////修改 2 处 依次修改为 private\protected
67.     {
68.     public:
69.         void showAge()
70.         {
71.             cout<<"the student age is"<<age<<endl;
72.         }
73.     };

```

结果:

```
30     student jack;
31     jack.age = 12;
32     jack.showAge();
33
34     return 0;
35 }
```

(9) 资源 编译日志 调试 搜索结果 关闭

单元	信息
D:\E盘迁移文件\中农二学位课程\2第二学期\CPP\代...	In member function 'void student::showAge()':
D:\E盘迁移文件\中农二学位课程\2第二学期\CPP\代...	[Error] 'int Human::age' is private
D:\E盘迁移文件\中农二学位课程\2第二学期\CPP\代...	[Error] within this context
D:\E盘迁移文件\中农二学位课程\2第二学期\CPP\代...	In member function 'void Primary::setAge(int)':
D:\E盘迁移文件\中农二学位课程\2第二学期\CPP\代...	[Error] 'int Human::age' is private
D:\E盘迁移文件\中农二学位课程\2第二学期\CPP\代...	[Error] within this context
D:\E盘迁移文件\中农二学位课程\2第二学期\CPP\代...	In function 'int main()':
D:\E盘迁移文件\中农二学位课程\2第二学期\CPP\代...	[Error] 'int Human::age' is private
D:\E盘迁移文件\中农二学位课程\2第二学期\CPP\代...	[Error] within this context

结论: 基类私有成员在派生类中不可见, 且不可被继承。

2.2 验证 protected 权限

2.2.1 protected 成员的公有继承

测试代码:

```
74.     class Human
75.     {
76.     protected://修改 1 处 依次修改为 private\protected
77.     int age;
78.     };
79.     class student:public Human////修改 2 处 依次修改为 private\protected
80.     {
81.     public:
82.     void showAge()
83.     {
84.     cout<<"the student age is"<<age<<endl;
85.     }
86.     };
87.     class Primary:public student////修改 3 处 依次修改为 private\protected
88.     {
```

```

88.     public :
89.         void setAge(int a)
90.         {age=a;}
        };

```

结果 1:

```

int main()
{
    Human man;
    man.age = 2;

    return 0;
}

```

资源 编译日志 调试 搜索结果 关闭

单元	信息
D:\E盘迁移文件\中农二学位课程\2第二学期\CPP\...	In function 'int main()':
D:\E盘迁移文件\中农二学位课程\2第二学期\CPP\代...	[Error] 'int Human::age' is protected
D:\E盘迁移文件\中农二学位课程\2第二学期\CPP\代...	[Error] within this context

结论 1: 保护的成员与私有成员一样，在类外不可见。

结果 2:

```

int main()
{
    student jack;
    // jack.age = 12;
    jack.showAge();
}

```

D:\E盘迁移文件\中农二学位课: × +

```

the student age is0
-----
Process exited after 0.7142 s

```

结论 2: 基类中的保护成员在派生类中可见;

结果 3:

```

int main()
{
    student jack;
    // jack.age = 12;
    jack.showAge();

    Primary jessic;
    // jessic.age=12;
    jessic.showAge();
    jessic.setAge(10);
    jessic.showAge();
    return 0;
}

```

```

D:\E盘迁移文件\中农二学位课: × +
the student age is0
the student age is0
the student age is10

-----
Process exited after 0.6902 se
请按任意键继续. . .

```

结论 3: 基类中的保护成员, 在派生类公有继承后访问权限为保护类型。

2.2.2 protected 成员的保护继承

测试代码:

```

91.  class Human
92.  {
93.      protected://修改 1 处 依次修改为 private\protected
94.      int age;
95.  };
96.  class student:protected Human////修改 2 处 依次修改为 private\protected
97.  {
98.      public:
99.      void showAge()
100.     {
101.         cout<<"the student age is"<<age<<endl;
102.     }
103.     };
104.  class Primary:public student////修改 3 处 依次修改为 private\protected
105.  {
106.      public :
107.      void setAge(int a)
108.      {age=a;}
109.      };

```

结果:

```

int main()
{
    student jack;
    // jack.age = 12;
    jack.showAge();

    Primary jessic;
    // jessic.age=12;
    jessic.showAge();
    jessic.setAge(10);
    jessic.showAge();
    return 0;
}

```

```

the student age is0
the student age is0
the student age is10

-----
Process exited after 0.70'
请按任意键继续. . .

```

结论：与 `protected` 成员的公有继承一样，在派生类中可见且在派生类中的访问权限为保护类型。

2.2.3 `protected` 成员的私有继承

测试代码：

```

108. class Human
109. {
110.     protected://修改 1 处 依次修改为 private\protected
111.     int age;
112. };
113. class student:private Human////修改 2 处 依次修改为 private\protected
114. {
115.     public:
116.     void showAge()
117.     {
118.         cout<<"the student age is"<<age<<endl;
119.     }
120. };
120. class Primary:public student////修改 3 处 依次修改为 private\protected
121. {
122.     public :
123.     void setAge(int a)
124.     {age=a;}
125. };

```

结果：


```

3  int main()
4  {
5      student jack;
6      // jack.age = 12;
7      jack.showAge();
8
9      Primary jessic;
0      // jessic.age=12;
1      jessic.showAge();
2      jessic.setAge(10);
3      jessic.showAge();
4      return 0;
5  }

```

资源	编译日志	调试	搜索结果	关闭
单元	信息			
D:\E盘迁移文件\中农二学位课程\2第二学期\CPP\...	In member function 'void Primary::setAge(int)':			
D:\E盘迁移文件\中农二学位课程\2第二学期\CPP\代...	[Error] 'int Human::age' is protected			
D:\E盘迁移文件\中农二学位课程\2第二学期\CPP\代...	[Error] within this context			

结论：基类中的保护成员在私有继承后，在派生类中不可见，且在派生类中的访问权限为私有权限。

2.3 保护继承

2.3.1 public 成员的保护继承

测试代码：

```

125.  class Human
126.  {
127.  Public://修改 1 处 依次修改为 private\protected
128.  int age;
129.  };
130.  class student:protected Human////修改 2 处 依次修改为 private\protected
131.  {
132.  public:
133.  void showAge()
134.  {
135.  cout<<"the student age is"<<age<<endl;
136.  }
137.  };
137.  class Primary:public student////修改 3 处 依次修改为 private\protected
138.  {
139.  public :

```



```

140. void setAge(int a)
141. {age=a;}
    };

```

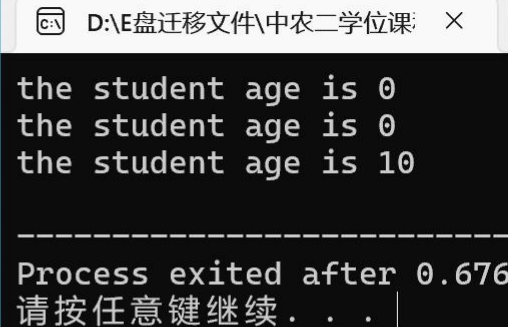
结果:

```

int main()
{
    student jack;
    // jack.age = 12;
    jack.showAge();

    Primary jessic;
    // jessic.age=12;
    jessic.showAge();
    jessic.setAge(10);
    jessic.showAge();
    return 0;
}

```



```

the student age is 0
the student age is 0
the student age is 10

-----
Process exited after 0.676
请按任意键继续...

```

结论: 基类公有成员经过保护继承后在派生类中可见, 且在派生类中访问权限为 **protected**。

2.3.2 protected 成员的保护继承

测试代码:

```

142. class Human
143. {
144. Protected://修改 1 处 依次修改为 private\protected
145. int age;
146. };
147. class student:protected Human////修改 2 处 依次修改为 private\protected
148. {
149. public:
150. void showAge()
151. {
152. cout<<"the student age is"<<age<<endl;
153. }
    };
154. class Primary:public student////修改 3 处 依次修改为 private\protected
155. {
156. public :
157. void setAge(int a)
158. {age=a;}
    };

```

结果:

```

int main()
{
    student jack;
    // jack.age = 12;
    jack.showAge();

    Primary jessic;
    // jessic.age=12;
    jessic.showAge();
    jessic.setAge(10);
    jessic.showAge();
    return 0;
}

```

```

the student age is 0
the student age is 0
the student age is 10

-----
Process exited after 0.68
请按任意键继续...

```

结论：基类保护成员经过保护继承后在派生类中可见，且在派生类中访问权限为 **protected**

三、总结

类有三种访问权限以及三种继承方式。

类的访问权限是针对类外与类内部的访问权限限制。在类的内部，不论何种访问权限，都可以被访问，在类的外部则需要遵循访问权限，只可以访问类的公有成员。

与之对应的继承方式也有三种，是针对类与类之间的继承方式而言的，基类不同权限的成员，经过不同方式的继承后会产生不同的新权限。

1. 基类的私有成员在派生类中不可见，不可以被继承；
2. 除基类私有成员外的任何成员，在经过私有继承后都在派生类中变为私有成员；
3. 公有继承不改变基类中的访问权限，在派生类中的权限与基类中一致；
4. 保护继承在继承基类公有成员时会将其改变为保护成员，而基类中的保护成员的权限在派生类中不变。