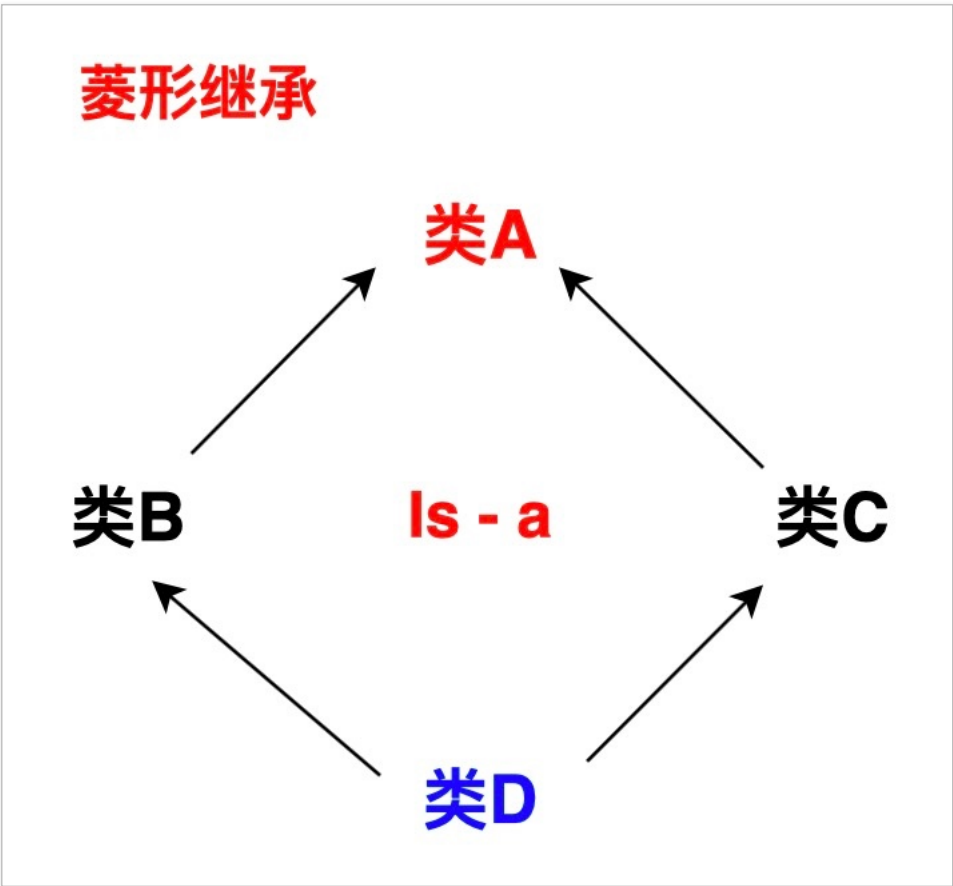


### 3.6 菱形继承\_物联网 / 嵌入式工程师 - 慕课网

“ 慕课网慕课教程 3.6 菱形继承涵盖海量编程基础技术教程，以图文图表的形式，把晦涩难懂的编程专业用语，以通俗易懂的方式呈现给用户。

6. 菱形继承



此时会导致 A 中的成员在类 D 中出现两次，产生歧义浪费空间

```
#include <iostream>

using namespace std;

class A{

public:
    int a;

};

class B:public A
{
```

```

};

class C:public A
{
};

class D:public B,public C
{
public:
    void function(void)
    {
        a = 200;
    }
};

int main(void)
{
    D object;
    object.a = 100;

    return 0;
}

```

程序运行结果如下:

```

linux@ubuntu:~/CPP_Class$ g++ Test.cpp
Test.cpp: In member function 'void D::function()':
Test.cpp:25:3: error: reference to 'a' is ambiguous
    a = 200;
    ^
Test.cpp:8:6: note: candidates are: int A::a
    int a;
    ^
Test.cpp:8:6: note:          int A::a
Test.cpp: In function 'int main()':
Test.cpp:32:9: error: request for member 'a' is ambiguous
    object.a = 100;
    ^
Test.cpp:8:6: note: candidates are: int A::a
    int a;
    ^
Test.cpp:8:6: note:          int A::a

```

解决方法，通过类名明确指定访问的是哪一个成员:

```
#include <iostream>

using namespace std;

class A{

public:
    int a;

};

class B:public A
{
};

class C:public A
{
};

class D:public B,public C
{
public:
    void function(void)
    {
        B::a = 200;
    }
};

int main(void)
{
    D object;
    object.C::a = 100;

    return 0;
}
```

- 划线
- 写笔记

学习要认真，笔记应当先



公开笔记 0/1000 提交



Sunny\_SunshineX

删除 编辑

全文完

本文由 简悦 SimpRead 优化，用以提升阅读体验  
使用了 全新的简悦词法分析引擎 beta，点击查看详细说明

