

原 C++练习题及答案（三）

2018年06月26日 21:45:03 快乐的小小程序猿 阅读数：1736

1

一、选择题

1. 下列类的定义中正确的是（ ）。
- ( A ) class a{int x=0;int y=1;} ( B ) class b{intx=0;int y=1;;}
- ( C ) class c{int x;inty;} ( D ) class d{int x;inty;;}
2. 若有以下说明，则在类外使用对象objX成员的正确语句是（ ）。
- class X
- { int a;
- voidfun1();
- public:
- voidfun2();
- };
- X objX;
- ( A ) objX.a=0; ( B ) objX.fun1(); ( C ) objX.fun2(); ( D ) X::fun1();
3. 在类定义的外部，可以被访问的成员有（ ）。
- ( A ) 所有类成员 ( B ) private或protected的类成员
- ( C ) public的类成员 ( D ) public或private的类成员
4. 下列关于类和对象的说法中，正确的是（ ）。
- ( A ) 编译器为每个类和类的对象分配内存 ( B ) 类的对象具有成员函数的副本
- ( C ) 类的成员函数由类来调用 ( D ) 编译器为每个对象的数据成员分配内存
5. 关于this指针的说法正确的是（ ）。
- ( A ) this指针必须显式说明 ( B ) 定义一个类后，this指针就指向该类
- ( C ) 成员函数拥有this指针 ( D ) 静态成员函数拥有this指针

【解答】 D D C D C

二、程序练习

1. 阅读程序，写出运行结果。

```
#include<iostream>

using namespacestd;

class A

{ public :

    int f1();

    int f2();
```

码书商店

谷歌开源移动UI框架Flutter学习必读出版社直供

关闭

```
void sety( int n ) { y =n; cout << y << endl; }

int getx() { returnx; }

int gety() { returny; }

private :

int x, y;

};

int A::f1()

{ return x + y; }

int A::f2()

{ return x - y; }

int main()

{ A a;

a.setx( 10 ); a.sety( 5 );

cout << a.getx() << '\t'<< a.gety() << endl;

cout << a.f1() << '\t' <<a.f2() << endl;

}
```

【解答】

2 . 改写以下程序。要求定义类student，封装三个数据成员和两个成员函数intpt和output，使程序得到相同的运行效果。

```
#include<iostream>

using namespacestd;

struct student

{ char name[20];

unsignedint id;

doublescore;

};

void input(student&stu)

{ cout<<"name?";

cin>>stu.name;

cout<<"id?";

cin>>stu.id;

cout<<"score?";

cin>>stu.score;

}

voidoutput(student &stu)

{ cout<<"name:"<<stu.name<<"\tid:"<<stu.id<<"\tscore: "<<stu.score<<endl; }

int main()
```

登录

注册

×

1

码书商店

谷歌  
开源  
移动  
UI  
框架  
Flutter  
学习  
必读

出版社  
直供

关闭

```
input(s);

output(s);

}

【解答】

#include <iostream>

using namespace std;

class student
{
    char name[20];

    unsigned int id;

    double score;

public:

    void input()
    {
        cout<<"name?";

        cin>>name;

        cout<<"id?";

        cin>>id;

        cout<<"score?";

        cin>>score;

    }

    void output()
    {
        cout<<"name:"<<name<<"\tid: "<<id<<"\tscore:"<<score<<endl;

    }

};

int main()
{
    student s;

    s.input();

    s.output();

}
```

1

同步练习6.2

一、选择题

- 1．下面对构造函数的不正确描述是（    ）。
- （ A ）用户定义的构造函数不是必须的

（ B ）构造函数可以重载

（ C ）构造函数可以有参数，也可以有返回值

（ D ）构造函数可以设置默认参数

码书商店

谷歌  
开源  
移动  
UI  
框架  
Flutter  
学习  
必读

出版社直供

关闭

2. 下面对析构函数的正确描述是（ ）。
- (A) 系统在任何情况下都能正确析构对象

(B) 用户必须定义类的析构函数

(C) 析构函数没有参数，也没有返回值

(D) 析构函数可以设置默认参数
3. 构造函数是在（ ）时执行的。
- (A) 建立源程序文件

(B) 创建对象

(C) 创建类

(D) 程序编译时
4. 在下列函数原型中，可以作为类Base析构函数的是（ ）。
- (A) void~Base

(B) ~Base()

(C) ~Base()const

(D) Base()
5. AB是一个类，那么执行语句 “ABa (4), b[3], \*p;” 调用了（ ）次构造函数。
- (A) 2

(B) 3

(C) 4

(D) 5
6. 下面关于复制构造函数调用的时机，不正确的是（ ）调用。
- (A) 访问对象时

(B) 对象初始化时

(C) 函数具有类类型传值参数时

(D) 函数返回类类型值时
7. 说明一个类的对象时，系统自动调用（ ）。
- (A) 成员函数

(B) 构造函数

(C) 析构函数

(D) 友元函数
8. 程序中撤销一个类对象时，系统自动调用（ ）。
- (A) 成员函数

(B) 构造函数

(C) 析构函数

(D) 友元函数

【解答】      C   C   B   B   C   A   B   C

## 二、程序练习

1. 阅读程序，写出运行结果。

```
#include<iostream>

using namespacestd;

class T
{ public :

    T( int x, int y )

    { a = x; b = y;

        cout << "调用构造函数1." << endl;

        cout << a << '\t' << b<< endl;

    }

    T( T &d )

    { cout << "调用构造函数2." << endl;

        cout << d.a << '\t' << d.b<< endl;

    }

    ~T() { cout << "调用析构函数."<<endl; }

    int add( int x, int y = 10 ) { returnx + y; }

private :

    int a, b;
```

1

码书商店

谷歌  
开源  
移动  
UI  
框架  
Flutter  
学习  
必读

出版社  
直供

关闭

```
};

int main()
{
    T d1( 4, 8 );

    T d2( d1 );

    cout << d2.add( 10 ) << endl;
}
```

【解答】

2 . 为同步练习6.1程序练习第2题中的student类增加一个构造函数，使得建立对象时可以完成用户指定数据的初始化。默认构造函数值为：( "\0", 0, 0 )

若主函数为：

```
int main()
{
    student s1;

    s1.output();

    student s2("Zhangsan", 120, 85);

    s2.output();

    student s3;

    s3.input();

    s3.output();
}
```

将有以下屏幕对话和输出：

```
name:      id: 0      score:0

name: Zhangsan      id:120      score: 85

name? Lihua

score? 95

name: Lihua      id: 130      score: 95
```

请补充student类的构造函数。

【解答】

```
class student
{
    char name[20];

    unsigned id;

    double score;

public:

    student(char s[20]="\0",unsigned k=0, double t=0)

    {

        strcpy_s(name,s);

        id=k;
```

1

码书商店



谷歌  
开源  
移动  
UI  
框架  
Flutter  
学习  
必读

出版社直供

关闭

```
        score=t;

    }

    void input()

    {

        cout<<"name?";

        cin>>name;

        cout<<"id?";

        cin>>id;

        cout<<"score?";

        cin>>score;

    }

    void output()

    {

        cout<<"name:"<<name<<"\tid: "<<id<<"\tscore:"<<score<<endl;

    }

};
```

一、选择题

1. 在下列选项中，（ ）不是类的成员函数。
- ( A ) 构造函数        ( B ) 析构函数        ( C ) 友元函数        ( D ) 复制构造函数
2. 下面对友元的错误描述是（ ）。
- ( A ) 关键字friend用于声明友元
- ( B ) 一个类中的成员函数可以是另一个类的友元
- ( C ) 友元函数访问对象的成员不受访问特性影响
- ( D ) 友元函数通过this指针访问对象成员
3. 已知类A是类B的友元，类B是类C的友元，则下面选项描述正确的是（ ）。
- ( A ) 类A一定是类C的友元
- ( B ) 类C一定是类A的友元
- ( C ) 类C的成员函数可以访问类B的对象的任何成员
- ( D ) 类A的成员函数可以访问类B的对象的任何成员
4. 下述关于类的静态成员的特性中，描述错误的是（ ）。
- ( A ) 说明静态数据成员时前边要加修饰符static
- ( B ) 静态数据成员要在类体外定义
- ( C ) 引用静态数据成员时，要在静态数据成员前加<类名>和作用域运算符
- ( D ) 每个对象有自己的静态数据成员副本
5. 若有以下说明，则对n的正确访问语句是（ ）。

```
class Y

{
    //...;
```

1

码书商店

谷歌  
开源  
移动  
UI  
框架  
Flutter  
学习  
必读

出版社直供

关闭

```
public:
    staticintn;

};

int Y::n;

Y objY;

( A ) n=1;          ( B ) Y::n=1;      ( C ) objY::n=1;      ( D ) Y->n
```

6 . 若有以下类Z说明，则函数fStatic中访问数据a错误的是（ ）。

```
class Z
{
    static int a;

public:
    staticvoid fStatic(Z&);

};

int Z::a=0; Z objZ;

( A ) voidZ::fStatic() { objZ.a =1; }
( B ) voidZ::fStatic() { a = 1; }
( C ) voidZ::fStatic() { this->a = 0; }
( D ) voidZ::fStatic() { Z::a = 0; }
```

7 . 若有以下类W说明，则函数fConst的正确定义是（ ）。

```
class W
{
    int a;

public:
    voidfConst(int&) const;

};

( A ) void W::fConst(int&k )const { k = a; }
( B ) void W::fConst(int&k )const { k = a++; }
( C ) void W::fConst(int&k )const { cin>> a; }
( D ) void W::fConst(int&k )const { a = k; }
```

8 . 若有以下类T说明，则函数fFriend的错误定义是（ ）。

```
class T
{
    inti;

    friendvoid fFriend( T&, int );

};

( A ) void fFriend( T&objT, int k ) { objT.i = k; }
( B ) void fFriend( T&objT, int k ) { k = objT.i; }
( C ) void T::fFriend( T&objT, int k ) { k += objT.i; }
( D ) void fFriend( T&objT, int k ) { objT.i += k; }
```

【解答】 C D D D B C A C

1

码书商店

谷歌  
开源  
移动  
UI  
框架  
Flutter  
学习  
必读

出版社  
直供

关闭

1

1

2

3

4

5

6

7

8

9

10

## 二、程序练习

1 . 阅读程序，写出运行结果。

```
#include<iostream>

using namespacestd;

class T
{ public:

    T(intx) { a=x; b+=x; };

    staticvoid display(T c) { cout<<"a="<<c.a<<"\t"<<"b="<<c.b<<endl; }

private:

    int a;

    staticint b;

};

int T::b=5;

int main()
{ T A(3), B(5);

  T::display(A);

  T::display(B);

}
```

【解答】

2 . 阅读程序，写出运行结果。

```
#include<iostream>

using namespacestd;

#include<cmath>

class Point
{ public :

    Point( float x, float y )

    { a= x; b = y; cout<<"点( "<<a<<" "<<b<<" )"; }

    friend double d( Point &A, Point&B )

    { return sqrt((A.a-B.a)*(A.a-B.a)+(A.b-B.b)*(A.b-B.b)); }

private:

    double a, b;

};

int main()

{ Point p1( 2, 3 );

  cout<<" 到";

}
```

码书商店



谷歌  
开源  
移动  
UI  
框架  
Flutter  
学习  
必读

出版社直供

关闭



```
Pointp2( 4, 5 );

cout<< "的距离是：" << d( p1,p2 ) << endl;

}
```

【解答】

3 . 阅读程序，写出运行结果。

```
#include<iostream>

using namespacestd;

class A
{ public :

    A() { a = 5; }

    void printa() { cout << "A:a =" << a << endl; }

private :

    int a;

    friend class B;

};

class B
{ public:

    voiddisplay1( A t )

    { t.a++; cout << "display1:a =" << t.a << endl; };

    voiddisplay2( A t )

    { t.a--; cout << "display2:a =" << t.a<< endl; };

};

int main()

{ A obj1;

  Bobj2;

  obj1.printa();

  obj2.display1(obj1 );

  obj2.display2(obj1 );

  obj1.printa();

}
```

【解答】

4 . 为同步练习6.2程序练习第2题中的student类添加一个复制构造函数。若主函数为：

```
int main()

{ cout<<"s2:\n";
```

1

码书商店

谷歌  
开源  
移动  
UI  
框架  
Flutter  
学习  
必读

出版社直供

关闭

```
student s2("Zhangsan", 120, 85);

s2.output();

cout<<"s3:\n";

student s3(s2);

s3.output();

}
```

则运行结果如下：

```
s2:
name: Zhangsan      id: 120    score: 85

s3:
name: Zhangsan      id: 120    score: 85
```

【解答】

```
class student
{
    char name[20];
    unsigned id;
    double score;
public:
    student(char s[20]= "\0", unsigned k=0, double t=0)
    {
        strcpy_s(name,s);
        id=k;
        score=t;
    }
    student(const student &ss)      //复制构造函数
    {
        strcpy_s(name,ss.name);
        id=ss.id;
        score=ss.score;
    }
    void input()
    {
        cout<<"name?";

        cin>>name;

        cout<<"id?";

        cin>>id;

        cout<<"score?";
```

1

码书商店

谷歌  
开源  
移动  
UI  
框架  
Flutter  
学习  
必读

出版社直供

关闭

```
        cin>>score;

    }

    void output()

    {

        cout<<"name:"<<name<<"\tid: "<<id<<"\tscore: "<<score<<endl;

    }

};
```

1

5 . 修改同步练习6.1程序练习第2题中的student类，把input和output函数写为友元函数，并相应修改主函数，使程序得到以下运行结果。

【解答】

```
#include<iostream>

#include<fstream>

using namespacestd;

class student

{

    char name[20];

    unsigned id;

    double score;

public:

    student(char s[20]="\0",unsigned k=0, double t=0)

    {

        strcpy_s(name,s);

        id=k;

        score=t;

    }

    student(const student &ss)

    {

        strcpy_s(name,ss.name);

        id=ss.id;

        score=ss.score;

    }

    friend void input(student&ss); //声明友元函数

    friend void output(student ss); //声明友元函数

};

void input(student&ss)

{

    cout<<"name? ";
```

码书商店

谷歌  
开源  
移动  
UI  
框架  
Flutter  
学习  
必读

出版社直供

关闭

```
cin>>ss.name;

cout<<"id? ";

cin>>ss.id;

cout<<"score? ";

cin>>ss.score;

}

voidoutput(student ss)

{

    cout<<"name:"<<ss.name<<"\tid:"<<ss.id<<"\tscore: "<<ss.score<<endl;

}

int main()

{

    student s1;

    input(s1);

    output(s1);

}
```

1

6．删除同步练习6.1程序练习第2题中student类的成员函数input和output，定义一个iostudent类，它是student类的友元类，完成对student数据成出操作。编写完整的程序，使其得到正确的运行效果。

【解答】

```
#include<iostream>

#include<fstream>

using namespacestd;

class student

{

    char name[20];

    unsigned id;

    double score;

public:

    student(char s[20]="\0",unsigned k=0, double t=0)

    {

        strcpy_s(name,s);

        id=k;

        score=t;

    }

    student(conststudent &ss)

    {

        strcpy_s(name,ss.name);
```

码书商店

谷歌  
开源  
移动  
UI  
框架  
Flutter  
学习  
必读

出版社直供

关闭

```
        id=ss.id;

        score=ss.score;

    }

    friend class iostudent;

};

class iostudent    //定义iostudent类
{
    public:

    void input(student &ss)

    {

        cout<<"name?";

        cin>>ss.name;

        cout<<"id?";

        cin>>ss.id;

        cout<<"score?";

        cin>>ss.score;

    }

    void output(student ss)

    {

        cout<<"name:"<<ss.name<<"\tid:"<<ss.id<<"\tscore: "<<ss.score<<endl;

    }

};

int main()

{

    student s1;

    iostudent io;

    io.input(s1);

    io.output(s1);

}
```

1

同步练习6.4

一、选择题

- 1 . 若classB中定义了一个classA的类成员A a , 则关于类成员的正确描述是 ( )。
- ( A ) 在类B的成员函数中可以访问A类的私有数据成员
- ( B ) 在类B的成员函数中可以访问A类的保护数据成员
- ( C ) 类B的构造函数可以调用类A的构造函数进行数据成员初始化
- ( D ) 类A的构造函数可以调用类B的构造函数进行数据成员初始化

码书商店



谷歌  
开源  
移动  
UI  
框架  
Flutter  
学习  
必读

出版社  
直供

关闭

2. 下列关于类的包含描述正确的是（ ）。

- ( A ) 可以使用赋值语句对对象成员进行初始化
- ( B ) 可以使用“参数初始式”调用成员类的构造函数初始化对象成员
- ( C ) 被包含类可以访问包含类的成员
- ( D ) 首先执行自身构造函数，再调用成员类的构造函数
- 【解答】 C B

1

二、程序练习

1. 阅读程序，写出运行结果。

```
#include<iostream>

using namespacestd;

class A
{ public:

    A(int x=0):a(x){ }

    void getA(int A) { a =A; }

    void printA() {cout<<"a="<<a<<endl; }

private:

    int a;

};

class B
{ public:

    B(int x=0, int y=0):aa(x) { b = y; }

    void getAB(int A, int outB) { aa.getA(A); b=outB; }

    void printAB() { aa.printA();cout<<"b="<<b<<endl; }

private:

    A aa;

    int b;

};

int main()
{ A objA;

    int m=5;

    objA.getA(m);

    cout<<"objA.a="<<m<<endl;

    cout<<"objB:\n";

    B objB;

    objB.getAB(12,56);

    objB.printAB();
```

码书商店



谷歌  
开源  
移动  
UI  
框架  
Flutter  
学习  
必读

出版社直供

关闭

}

【解答】

2. 为同步练习6.1程序练习第2题中的student类添加一个date类数据成员birthday，date类包含三个数据成员：year、month、day，以及用于初始化的数据成员，用于输入数据的input和输出数据的output成员函数。student类构造函数需要完成birthday的数据初始化，并且完成birthday成员函数的输入/输出。测试这个类。

【解答】

```
#include <iostream>

#include<fstream>

using namespace std;

class date //定义date类
{
    int year, month, day;

public:
    date(int y, int m, int d)
    {
        year=y;
        month=m;
        day=d;
    }

    void input()
    {
        cout<<"birth of year ? ";
        cin>>year;

        cout<<"\t month? ";
        cin>>month;

        cout<<"\t day ? ";
        cin>>day;
    }

    void output()
    {
        cout<<"birth:"<<year<<"-"<<month<<"-"<<day<<endl;
    }

};

class student //定义student类
{
    char name[20];
    unsigned id;
```

码书商店

谷歌  
开源  
移动  
UI  
框架  
Flutter  
学习  
必读

出版社直供

关闭

```
double score;

date birth;    //date类的数据成员

public:

    //构造函数

    student(chars[20]="No name", unsigned k=0, double t=0, int y=2000, int m=1, int d=1)

: birth(y, m, d)

{

    strcpy_s(name,s);

    id=k;

    score=t;

}

void input()

{

    cout<<"name?";

    cin>>name;

    birth.input();

    cout<<"id?";

    cin>>id;

    cout<<"score?";

    cin>>score;

}

void output()

{

    cout<<"name:"<<name<<"\t";

    birth.output();

    cout<<"id:"<<id<<"\tscore: "<<score<<endl;

}

};

int main()

{

    students;

    s.input();

    s.output();

}
```

综合练习

一、思考题

1．结构与类有什么区别？如果把程序中定义结构的关键字struct直接改成class，会有什么问题？用教材中的一个例程试试看，想一想做什么修改能使用？

码书商店



谷歌  
开源  
移动  
UI  
框架  
Flutter  
学习  
必读

出版社直供

关闭



【解答】

结构是数据的封装，类是数据和操作的封装。可以把结构看成是类的特例。结构和类都可以用关键字struct或class定义。区别是：struct定义的结构或都是公有的，用class定义的结构或类不做声明的成员是私有的。

若把struct改成class，只需要把全部成员定义为public就可以了。

2. 有说明：

```
class A
{
    int a;

    double x;

public:
    funMember();
};

A a1, a2, a3;
```

编译器为对象a1、a2和a3开辟了什么内存空间？它们有各自的funMember函数的副本吗？C++通过什么机制调用类的成员函数？

【解答】

开辟的存储空间有a1.a,a1.x, a2.a, a2.x, a3.a, a3.x。各对象没有funMember函数的副本，C++通过this指针调用成员函数。

3. C++提供了系统版本的构造函数，为什么还需要用户自定义构造函数？编写一个验证程序，说明自定义构造函数的必要性。

【解答】

类的默认构造函数可以建立基本类型数据成员的存储空间。基于以下两个原因，需要用户定义构造函数：

- （1）对数据成员的值做指定初始化；
- （2）类的数据是由指针管理的堆。

程序略。

4. 试从定义方式、访问方式、存储性质和作用域4个方面来分析类的一般数据成员和静态数据成员的区别，并编写一个简单程序验证它。

【解答】

	定义方式	访问方式	存储性质	作用域
一般数据成员	类中定义	对象.数据成员	局部数据	由访问属性public, protected, private决定
静态数据成员	类中声明，类外定义	对象.数据成员 类::数据成员	全局数据	

程序略。

5. 试从定义方式、调用方式两个方面来分析常成员函数、静态成员函数和友元函数的区别。考察例6-16，若class Goods的指针域：

```
Goods * next;
```

被声明为私有（private）成员，程序会出现什么错误？做什么最小修改能使程序正确运行？

【解答】

	定义方式	调用方式
--	------	------

码书商店

谷歌  
开源  
移动  
UI  
框架  
Flutter  
学习  
必读

出版社  
直供

关闭

常成员函数	函数原型以const做后缀 this指针被约束为指向常量的常指针	与一般成员函数调用形式相同 对数据成员只读	1	
静态成员函数	以static做函数原型前缀 没有this指针	通过类或对象调用 用于操作静态数据成员		
友员函数	以friend做函数原型前缀 没有this指针	通过参数访问对象 可以访问对象的不同属性		

在例6-16中，若把next声明为私有数据成员，只须把有关指针操作的函数定义为友员函数就可以了：

```
friend void purchase( Goods * &f, Goods * &r, int w );  
  
friend void sale( Goods * &f ,Goods * &r );
```

6 . 设有：

```
class M  
{ public: int a;  
  
};  
  
class N  
{ public:  
  
    M m;  
  
    int b;  
  
    void fun()  
    { /*...*/ }  
  
};  
  
int main()  
{ N n;  
  
    N *p = &n;  
  
    /*...*/  
  
}
```

描述在N::fun中如何访问M类的数据成员a？在main函数中又如何访问对象n的全部数据成员？

【解答】

在N::fun中访问M类的数据成员a的形式是： m.a

在main函数中访问M类的数据成员的形式是： n.b，n.m.a

## 二、程序设计

1 . 定义一个Book（图书）类，在该类定义中包括以下数据成员和成员函数。

数据成员：bookname（书名）、price（价格）和number（存书数量）。

成员函数：display()显示图书的情况；borrow()将存书数量减1，并显示当前存书数量；restore()将存书数量加1，并显示当前存书数量。

在main函数中，要求创建某一种图书对象，并对该图书进行简单的显示、借阅和归还管理。

【解答】

```
#include <iostream>
```

码书商店

谷歌  
开源  
移动  
UI  
框架  
Flutter  
学习  
必读

出版社  
直供

关闭

```
using namespace std;

classBook
{
public:
    void setBook(char*,double,int);

    void borrow();

    void restore();

    void display();

private:
    char bookname[40];

    double price;

    intnumber;

};

//在类外定义Book类的成员函数

voidBook::setBook(char *name, double pri, int num)
{
    strcpy_s(bookname, name);

    price=pri;

    number=num;
}

voidBook::borrow()
{
    if (number==0 )
    {
        cout<< "已没存书，退出！"<< endl;

        abort();
    }

    number = number - 1;

    cout << "借一次，现存书量为："<< number << endl;
}

voidBook::restore()
{
    number = number + 1;

    cout << "还一次，现存书量为："<< number << endl;
}

voidBook::display()
{

```

1

码书商店



谷歌  
开源  
移动  
UI  
框架  
Flutter  
学习  
必读

出版社  
直供

关闭

```
cout<< "存书情况："<< endl

    << "bookname:" <<bookname << endl

    << "price:" <<price << endl

    << "number:" <<number << endl;

}

intmain()

{

    char flag, ch;

    Book computer;

    computer.setBook( "C++程序设计基础", 32, 1000 );

    computer.display();

    ch = 'y';

    while ( ch == 'y' )

    {

        cout << "请输入借阅或归还标志(b/r)：" ;

        cin >> flag;

        switch ( flag )

        {

            case 'b': computer.borrow(); break;

            case 'r': computer.restore();

        }

        cout << "是否继续？(y/n)";

        cin>> ch;

    }

    computer.display();

}
```

1

2 . 定义一个Box（盒子）类，在该类定义中包括以下数据成员和成员函数。

数据成员：length（长）、width（宽）和height（高）。

成员函数：构造函数Box，设置盒子的长、宽和高三个初始数据；成员函数setBox对数据成员置值；成员函数volume计算盒子的体积。

在main函数中，要求创建Box对象，输入长、宽、高，输出盒子的体积。

【解答】

```
#include<iostream>

using namespacestd;

class BOX

{

    public:
```

码书商店



谷歌  
开源  
移动  
UI  
框架  
Flutter  
学习  
必读

出版社直供

关闭

```
BOX( double l, double w, double h )

{ length = l;

    width = w;

    height = h;

}

void volume()

{ cout << "volume=" << length * width * height<< endl;

}

private:

    double length, width, height;

};

int main()

{

BOX box1( 1,3,5 );

box1.volume();

BOX box2( 2,4,6 );

box2.volume();

}
```

1

3 . 定义一个student类，在该类定义中包括：一个数据成员（分数score）及两个静态数据成员（总分total和学生人数count）；成员函数scoretotalcount(doubles) 用于设置分数、求总分和累计学生人数；静态成员函数sum()用于返回总分；静态成员函数average()用于求平均值。

在main函数中，输入某班同学的成绩，并调用上述函数求全班学生的总分和平均分。

【解答】

```
#include <iostream>

using namespace std;

classstudent

{

public:

voidscoretotalcount( double s )

{

score= s;

    total= total + score;

    count++;

}

staticdouble sum()

{

    return total;

}
```

码书商店

谷歌  
开源  
移动  
UI  
框架  
Flutter  
学习  
必读

出版社直供

关闭

```
staticdouble average()

{

    return total / count;

}

private:

double score;

staticdouble total;

staticdouble count;

};

doublestudent::total=0;

doublestudent::count=0;

intmain()

{

    int i,n; double s;

    cout << "请输入学生人数：";

    cin >> n;

    student stu;

    for( i=1; i<=n; i++ )

    {

        cout << "请输入第"<< i << "个学生的分数：";

        cin >> s;

        stu.scoretotalcount( s );

        cout << "总分： "<< student::sum() << endl;

    }

    cout<< "平均分： "<< student::average() << endl;

}
```

4．定义一个表示点的结构类型Point和一个由直线方程 $y = ax + b$ 确定的直线类Line。结构类型Point有两个成员x和y，分别表示点的横坐标和纵坐标。Line类有两个数据成员a和b，分别表示直线方程中的系数。Line类有一个成员函数print用于显示直线方程。友元函数setPoint(Line&l1,Line &l2)用于求两条直线的交点。在main函数中，建立两个直线对象，分别调用print函数显示两条直线的方程，并调用函数setPoint求这两条直线的交点。

【解答】

```
#include <iostream>

using namespace std;

structpoint

{

    double x; double y;

};

classline

{
```

1

码书商店

谷歌开源移动UI框架Flutter学习必读

出版社直供

关闭

```
public:

    line(double u, double v )

    {

a=u; b=v;

    }

    void print()

    {

cout<<"y="<<a<<"x+"<<b<<endl;

    }

    friend point setpoint(line &l1,line &l2);

private:

    double a, b;

};

point setpoint( line &l1, line &l2 )

{

point p;

p.x=(l2.b-l1.b)/( l1.a-l2.a );

p.y=(l1.a*l2.b-l2.a*l1.b)/(l1.a-l2.a );

return p;

}

int main()

{

point setp;

    line l1(2,3), l2(4,5);

    cout<<"直线l1:";

    l1.print();

    cout<<"直线l2:";

    l2.print();

    setp=setpoint( l1,l2 );

    cout<<"直线l1和直线l2的交点:("<<setp.x<<","<<setp.y<<")"<<endl;

}


```

1

5．用类成员结构修改第4题的程序，使其实现相同的功能。定义Point类和Line类，表示点和线；定义setPoint类，包含两个Line类成员和一个表示直线Point成员，并定义类中求直线交点的成员函数。编写每个类相应的成员函数和测试用的主函数。

【解答】  
略。

码书商店




谷歌  
开源  
移动  
UI  
框架  
Flutter  
学习  
必读

出版社直供

关闭

股市彻底变天了，不看你就亏大了！

禄基 · 顶新

		1	
	想对作者说点什么		
<b>C++练习题及答案（一）</b>	第4章练习题同步练习4.1一、选择题1. 有数组定义 doubled[10]; 以下叙述不正确的是（ ）。( A ) 数组d有10个...	博文	来自： 快乐的序猿
<b>C++经典习题</b>	1、 设计一个立方体类BOX，它能计算并输出立方体的体积和表面积。 提示：定义一个BOX类，含有一个私有数据...	博文	来自： qq_20...的博客
<b>经典C++笔试题目--100(编程练习(比C要难)(91-100))</b>	编程练习(比C要难)(91-100)91、请编写一个C函数，该函数给出一个字节中被置1的位的个数。【参考答案】unsign...	博文	来自： xiangji...

股市彻底变天了，不看你就亏大了！

禄基 · 顶新

<b>C/C++题库整理（一）</b>	2017年10月1日牛客网题目一：/*小Q最近遇到了一个难题：把一个字符串的大写字母放到字符串的后面，各个字符...	博文	来自： Sonihr的犀牛园	阅读数 1
<b>C++（初学者）练习题-day one</b>	小编提示： 下面的程序代码用C++程序实现，适合初学者参考学习。同样的，我在写程序的过程中，会注释一些个...	博文	来自： weizu_cool	阅读数 766
<b>C++经典练习题合集</b>	经典的C++练习题，大家可以试试，特别是对于那些刚刚入门学习C++的同学，值得一试			09-03 下载
<b>C++算法习题</b>	1.分别写出bool,int,float,指针类型变量与零的比较2.判断一段程序是由c还是由c++编写3.递归反向输出字符串4.用递...	博文	来自： 大野狼来啦	阅读数 347
<b>C++（例题集—简单解析-类与对象）</b>	由于自己本人上课不太认真，只能靠自己学习例题总结。希望能给大家一些便捷，让大家开心学习C++。对象的创建...	博文	来自： Z_sea的博客	阅读数 1814
<b>良乡25岁美女手机做这个，1年存款吓呆父母！！</b>				
<b>精选精搜C++实例30道必背经典+100道题库</b>	压缩包内有工程代码30个和100道文档题库			07-30 下载
<b>C++ 练习问题</b>	学习C++ 一段时间了，请问在哪里有大量的基础习题来操作一下巩固一下的，不胜感激			论坛
 <b>671coder</b> 166篇文章 排名:2000+	 <b>50no</b> 8篇文章 排名:千里之外	 <b>drivedreams</b> 6篇文章 排名:千里之外	 <b>IAMoldpan</b> 115篇文章 排名:千里之外	
<b>C++ Primer第五版中文版习题集(保证完整版)</b>	学习最好的方法就是练习编写程序,本习题同步c++ Primer 第5 中文版			11-02 下载
<b>C++ 例题</b>	1、C++与C的区别？C是面对过程的，C++是面对对象的，C是一个结构化语言，它的重点在于算法和数据结构。C...	博文	来自： Jang1996的博客	阅读数 247
<b>c++练习题，简单得很</b>	好资料c++练习题，需要的就下载下来吧，很好的资源			03-07 下载

长喝蜂蜜的人和不喝蜂蜜的人真的有区别，后悔没早知道！！

舒恒 · 鸚鵡



<b>C++ Primer习题集 第5版-Stanley B. Lippman_高清版[带完整书签目录]</b>	03-29
C++ Primer习题集 第5版-Stanley B. Lippman_高清版 【作 者】（美）李普曼，（美）拉乔伊，（美）默著 【出版发行】北京：电子工业出...	下载
<b>C++之基础练习题</b>	1 数 1674
C与C++的差异【题目1】C和C++中struct有什么区别？【参考答案】Protection行为能否定义函数C无不能，但可... 博文 来自： Lewisg	
<b>各个软件公司+面试题+笔试题+c+c%2B%2B有答案.rar</b>	
各个软件公司+面试题+笔试题+c+c%2B%2B有答案.rar	下载
<b>c%2B%2B语言程序设计课后答案(清华大学郑莉).rar</b>	
这是 清华大学郑莉老师所编辑的 c语言程序设计课后答案 ... 下载时 请看清楚！	下载
<b>50道c++编程练习题</b>	
50道经典习题，简介但不失经典，为c++初学者提供一定的指导，很值得学习	下载
<b>股票“三不卖七不买”6字黄金口诀，高手教你一个涨停铁定律</b>	
禄基·顶新	
<b>C++函数基础习题及答案</b>	阅读数 873
1、编写一个求x的n次方的函数，并用它求5的3次方的值，用main()调用。2、输入一个8位二进制数，将其转换为十... 博文 来自： koko_pe的博客	
<b>c++教程课后练习题</b>	阅读数 504
1：在一个循环中读入两个int并打印出来，如果遇到‘ ’就退出程序。分析：从键盘输入中读取两个数，有可能读的... 博文 来自： bai_lu_sz的专栏	
<b>接口测试该怎么做</b>	阅读数 64
结合工作实际和学习其他人的总结，是时候对“接口测试该怎么做”来一个梳理了。一、对于接口测试来说，项目测... 博文 来自： heimao0307的博客	
<b>C++类(Class)的定义与实现</b>	阅读数 726
一、&quot;类&quot;的介绍 在C++中,用&quot;类&quot;来描述&quot;对象&quot;,所谓的&quot;对象&quot;,... 博文 来自： xulingxin的博客	
<b>接口测试总述</b>	阅读数 31
脱离页面呈现，前段页面尚未完成，直接测试接口功能。重点是检查数据的交换、传递的正确性。通常包括测试接... 博文 来自： jdhfjb的专栏	
<b>回忆小时候的味道，秦岭山脉正宗土蜂蜜！快来抢购吧！</b>	
至诚家·顶新	
<b>接口测试</b>	阅读数 138
一、接口测试的基本定义 接口测试是测试系统组件间接口的一种测试。接口测试主要用于检测外部系统与系统之间... 博文 来自： waiting666的博客	
<b>C语言循环结构练习题</b>	11-18 下载
C语言循环结构练习题	
<b>C语言程序设计现代方法第二版，第三章课后编程习题全部答案</b>	阅读数 974
小白一只，可能会有错误3.1#include<stdio.h>;intmain(void){ intday,month,year; printf(&quot;... 博文 来自： 编程初学者学习笔记	
<b>c语言编程练习题及答案</b>	阅读数 7777
【程序1】题目：有1、2、3、4个数字，能组成多少个互不相同且无重复数字的三位数？都是多少？1.程序分析：可... 博文 来自： drivedreams的专栏	
<b>C与指针课后练习与编程答案（不断更新）</b>	阅读数 609
不知第几遍看C与指针了，这么一本不厚的书却感觉将C语言的精髓讲了出来，这本书或许不是最适合入门的C语言， ... 博文 来自： OLDPAN的博客	
<b>养蜂人教你如何正确区分“真假”蜂蜜！</b>	
至诚家·顶新	
<b>C语言练习题（带详解答案）</b>	04-30 下载
C语言练习题（带详解答案）想学习C语言的朋友可以有很多用处	

码书商店

谷歌  
开源  
移动  
UI  
框架  
Flutter  
学习  
必读

出版社直供


关闭

<b>C++一些经典例题</b>	阅读数 127
C++一些经典例题1、求一个数的绝对值#includeusingnamespacestd;intmain(){intx,y;cin&amp;amp;gt;&a... 博文 来自： lyr201717的...	
<b>C++复习题集</b>	1 11-01
刚开始学习c++时感觉很困难，尤其是对于里面的一些语法，更是感觉很懵。直接做题感觉是进步最快的方法。	下载
<b>c/c++ 基本习题练习</b>	数 1458
求从【n,m】的和编写函数求【n,m】的整数的和；求绝对值函数编写代码，求一个整数的绝对值判断成绩等级计算... 博文 来自： Econoark的...	
<b>C++练习题及答案（四）</b>	数 523
第7章练习题同步练习7.1一、选择题1. 在下列运算符中，不能重载的是（ ）。（A）！（B）sizeof ... 博文 来自： 快乐的小程序猿	

**afp金融理财师相关资讯！有些人还不知道，赶快来看看！**

华金金考·顶新

<b>C++友元函数基础练习</b>	阅读数 235
#include#includeusingnamespacestd;/**定义Time类*数据成员：m_iHour，m_iMinute，m_iSecond*成员函数... 博文 来自： Steve_Abelieve	
<b>C++（笔记）友元函数、const基础练习</b>	阅读数 612
/*定义一个类Cat，包含一个私有成员变量weight，设置好weight的存取函数。定义一个类Dog，包含一个私有成员... 博文 来自： 浅然的专栏	
<b>C++ 习题 输出日期时间--友元函数</b>	阅读数 916
Description设计一个日期类和时间类，编写display函数用于显示日期和时间。要求：display函数作为类外的普通函... 博文 来自： so_so_y的博客	
<b>C++试题</b>	阅读数 86
一、简答题。1.voidGetMemory(char**p,intnum){*p=(char*)malloc(num);} voidTest(void){char*str=NULL; ... 博文 来自： lgy101024的博客	
<b>C++编程试题</b>	04-09
C++编程试题	下载



**对于程序员来说，英语到底多重要？**

不背单词和语法，一个公式秒懂英语！

<b>C++习题答案</b>	下载
周霁如和林伟健 版本C++程序设计基础（第五版）（上）	
<b>vc%2B%2B上位机串口程序(实时曲线功能).RAR</b>	下载
vc%2B%2B上位机串口程序(实时曲线功能).RAR	
<b>CCF中学生计算机程序设计入门篇练习2.4.2(NOI 1002 三角形) pascal</b>	阅读数 1105
1002.三角形(StandardIO)题目描述输入三角形三边长a,b,c（保证能构成三角形），输出三角形面积。... 博文 来自： qq_15851711的博客	
<b>c语言数组练习题及答案</b>	12-16
这东西对基础不扎实的同学最好，基础好的也可以来看看	下载
<b>C语言指针相关定义的练习题及答案</b>	阅读数 1310
C语言指针定义方面几点练习 博文 来自： 窗外阴天的博客	



**金融女海归,带你3个月入门机器学习！**

用最简单的语言，把算法讲的有深度！菜菜的机器学习sklearn课堂，立抢席位

<b>C语言练习题及答案</b>	10-16
计算机C语言考试练习题，复习题，非常实用。	下载
<b>C语言第五章习题参考答案</b>	12-10
C语言第五章习题参考答案，欢迎下载,仅供参考！	下载

[C++] 经典练习题与解析（二）

( 11 ) #include struct A{A(){std::cout

C++习题练习

该习题是基础的C++练习题，也是考试常考的内容，有求最大公约数和最小公倍数，因子，计算阶乘，求和，求数列等等

C++经典练习例题200例

这个是由浅到深的学习C++的 感觉很好 txt格式的



50万码农评论：英语对于程序员有多重要！

不背单词和语法，老司机教你一个数学公式秒懂天下英语

C++练习题及答案（二）

一、选择题1 . 语句 cout&lt;&lt;(1&amp;2)&lt;&lt;"&lt;&lt;(1...

C++习题 矩阵求和--重载运算符

/\*Description有两个矩阵a和b，均为2行3列。求两个矩阵之和。重载运算符“+”，使之能用于矩阵相加（如c=a+...

eNSP V390 USG6000v WEB配置（史上最详细）

看一下本帅帅的硬件环境 win7 64 位企业版 查看下我的VirtualBox版本 看下 我下载的文件和解压出来的文件 拉出一...

caffe安装系列——安装OpenCV

说明 网上关于caffe的安装教程非常多，但是关于每一步是否操作成功，出现了什么样的错误又该如何处理没有给出...

【Unity3D Shader编程】之二 雪山飞狐篇：Unity的基本Shader框架写法&颜色、光照与材质

本篇文章中，我们学习了Unity Shader的基本写法框架，以及学习了Shader中Properties（属性）的详细写法，光照...

Axure RP 8 注册码(最新亲测可用)

Axure RP 8.0 注册码仅供个人学习交流使用（建议购买正版授权）8.1.0.3366亲测可用 Licensee：University of Sci...

QT选择目录等常用文件/文件夹操作

QT 创建文件夹 bool QDir::mkdir ( const QString & dirName ) const 创建一个子目录名为目录名。[喝小酒...

两个重叠模型模拟双pass,顺便解决透明乱序问题

虚幻4除了能用rendertarget模拟双pass行为，还可以用两个模型来模拟双pass。用两个一模一样的模型，第一个模...

Android 增强版百分比布局库 为了适配而扩展

转载请标明出处：http://blog.csdn.net/lmj623565791/article/details/46767825； 本文出自：【张鸿洋的博客】 ...

【机器学习】Logistic Regression 的前世今生（理论篇）

Logistic Regression可以说是机器学习的入门算法。不过，你真的有把握能够把LR从头到脚讲一遍吗？你会现场建模...

C#实现开发windows服务实现自动从FTP服务器下载文件（自行设置分/时执行）

最近在做一个每天定点从FTP自动下载节目.xml并更新到数据库的功能。首先想到用 FileSystemWatcher来监控下载...

jquery/js实现一个网页同时调用多个倒计时(最新的)

jquery/js实现一个网页同时调用多个倒计时(最新的) 最近需要网页添加多个倒计时. 查阅网络,基本上都是千篇一律的...

ThreadLocal的设计理念与作用

Java中的ThreadLocal类允许我们创建只能被同一个线程读写的变量。因此，如果一段代码含有一个ThreadLocal变...

《联众》并购案细节公布，海虹海外公司浮出水面(ZT)

《联众》一夜身价暴增到2亿美金，《海虹》以及联众创始人成功套现1亿美金！这是中韩国际资本合作的典范？...

WebService学习（二）—— 调用自定义对象参数

WebService学习（二）—— 调用自定义对象参数 本文主要内容： 1、如何通过idea进行WebService Client的简单...

SNMP协议详解<二>

上一篇文章讲解了SNMP的基本架构，本篇文章将重点分析SNMP报文，并对不同版本（SNMPv1、v2c、v3）进行...

阅读数 4140

博文 来自： (ง •̀\_•́)ง

1 08-20

下载

05-15

下载

数 695

博文 来自： 快乐的小小程序猿

阅读数 2799

博文 来自： u012369559的专栏

阅读数 3万+

博文 来自： Kilven

阅读数 1万+

博文 来自： 张学志的博客

阅读数 5万+

博文 来自： 【浅墨的游戏编程B...

阅读数 11万+

博文 来自： 前端大白兔的博客

阅读数 1万+

博文 来自： K7的专栏

阅读数 3473

博文 来自： qq\_16756235的博客

阅读数 13万+

博文 来自： Hongyang

阅读数 2万+

博文 来自： AutoVision (by 仙...

阅读数 1万+

博文 来自： kongwei521的专栏

阅读数 21万+

博文 来自： websites

阅读数 2万+

博文 来自： u011860731的专栏

阅读数 5870

博文 来自： 曾登高

阅读数 1万+

博文 来自： 止水的专栏

阅读数 3万+

博文 来自： 假装在纽约

码书商店

谷歌开源移动UI框架Flutter学习必读

出版社直供

关闭

<b>基于Linux的智能家居的设计(5)</b>	1万+
4 软件设计 软件设计部分主要包括uboot移植、内核编译、系统移植、设备驱动编程、应用程序编程（QT编程、m... 博文 来自： 求是07	
<b>python图片处理类之~PIL.Image模块(ios android icon图标自动生成处理)</b>	2万+
1.从pyCharm提示下载PIL包 http://www.pythonware.com/products/pil/ 2.解压后，进入到目录下 cd /Users/ji... 博文 来自： 专注于 +unit...	
<b>centos 查看命令源码</b>	3万+
# yum install yum-utils 设置源: [base-src] name=CentOS-5.4 - Base src - baseurl=http://vault.ce... 博文 来自： linux/	
<b>SSM框架——详细整合教程（Spring+SpringMVC+MyBatis）</b>	45万+
使用SSM（Spring、SpringMVC和Mybatis）已经有三个多月了，项目在技术上已经没有什么难点了，基于现有的... 博文 来自： 在路上	
<b>强连通分量及缩点tarjan算法解析</b>	35万+
强连通分量：简言之 就是找环（每条边只走一次，两两可达）孤立的一个点也是一个连通分量 使用tarjan算法 在... 博文 来自： 九野的	
<b>史上最好的LDA(线性判别分析)教程</b>	2万+
一、前言最近由于研究需要，要用到线性判别分析(LDA)。于是找了很多资料来看，结果发现大部分讲的都是理论知... 博文 来自： jnulzl的专栏	
<b>CCS6.2超详细使用方法</b>	3万+
CCS6.2超详细使用方法 本文介绍了如何安装CCS6.2、一步一步的建立工程、以及建立工程以后编译、调试、如何方... 博文 来自： 火山LF的博客	
<b>maven项目生成的war包在tomcat下运行报错</b>	1万+
maven项目在tomcat 博文 来自： 直到世界的尽头	
<b>Mysql 查询一天中每半小时记录的数量</b>	5539
SELECT HOUR(e.time)as Hour,FLOOR(MINUTE(e.time)/30) as M, COUNT(*) as Count FROM error_log e WHE... 博文 来自： 刘宇(LY)个人笔记	
<b>Python(2) 基础语法</b>	4176
1. 模块1.1. 从某模块导入函数import somemodule from somemodule import somefunction from somemodul... 博文 来自： 清欢	
<b>关于计算时间复杂度和空间复杂度</b>	4万+
相信学习编程的同学，或多或少都接触到算法的时间复杂度和空间复杂度了，那我来讲讲怎么计算。 常用的算法... 博文 来自： 杨威的博客	
<b>R语言逻辑回归、ROC曲线和十折交叉验证</b>	3万+
自己整理编写的逻辑回归模板，作为学习笔记记录分享。数据集用的是14个自变量Xi，一个因变量Y的australian数据... 博文 来自： Tiaaaaaa的博客	
<b>web.config中的session配置详解</b>	1万+
打开某个应用程序的配置文件Web.config后，我们会发现以下这段： < sessionState mode="InProc" stateCo... 博文 来自： yszwn的专栏	
<b>关于SpringBoot bean无法注入的问题（与文件包位置有关）</b>	10万+
问题场景描述整个项目通过Maven构建，大致结构如下： 核心Spring框架一个module spring-boot-base service... 博文 来自： 开发随笔	
<b>DirectX修复工具增强版</b>	183万+
最后更新：2018-12-20 DirectX修复工具最新版：DirectX Repair V3.8 增强版 NEW! 版本号：V3.8.0.11638 大小： ... 博文 来自： VBcom的专栏	
<b>【机器学习】【层次聚类算法-1】HCA(Hierarchical Clustering Alg)的原理讲解 + 示例展示数...</b>	790
1.聚类系列算---层次聚类算法2.层次聚类算法的计算原理3.一个示例展示层次聚类算法的数学计算过程参考文献：[1]... 博文 来自： 蔚蓝的天空Tom	
习题答案 C/C++ C/C++培训 C/C++培训套餐 C/C++入门教程	
c++章末练习题 c++标准模板库编程实战答案 c++实验2答案 c++拷贝构造函数练习题 c++编程练习题 python教程答案 人工智能教程课后答案	



快乐的小小程序猿

关注

原创

91

粉丝

53

喜欢

27

评论

11

码书商店



谷歌开源移动UI框架Flutter学习必读

出版社直供

关闭

等级：

博客 4

访问：3万+

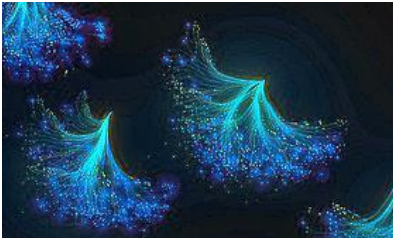
积分：1229

排名：5万+


勋章：

恒

1024



数据 可视化



最新文章

网络安全-练习RSA算法

网络安全-古典加密算法之置换密码加解密算法

网络安全-古典加密算法之替代密码加解密算法

网络安全-利用sniffer工具，分析TCP头的结构，并分析TCP的三次握手过程（详细）

网络安全-利用sniffer抓包软件，分析IP头的结构（详解）

归档

2019年2月9篇

2019年1月19篇

2018年12月7篇

2018年11月8篇

2018年10月7篇

展开

热门文章

C++练习题及答案（一）  
阅读数 9304

找7的倍数和含有7的数问题  
阅读数 2925

LINUX安装配置（详细）  
阅读数 2248

基于Java Web的网上图书商城管理系统——（一）  
阅读数 1911

C++练习题及答案（三）  
阅读数 1731

最新评论

基于Java Web的网上图书商城...  
xu\_benjamin：慢慢来，总会解决的

基于Java Web的网上图书商城...  
wjkdxdjq：真的是太棒了，哇，我们实训就是做一个当当网的购物程序，完全不知道该怎么做啊

基于Java Web的网上图书商城...  
xu\_benjamin：[reply]weixin\_43025559[/reply]  
资料+源码 链接：https://p...

基于Java Web的网上图书商城...

1

码书商店



谷歌  
开源  
移动  
UI  
框架  
Flutter  
学习  
必读

出版社  
直供

关闭

xu\_benjamin：好的，抽空把一些文件放到百度网盘上

[基于Java Web的网上图书商城...](#)

weixin\_43025559：麻烦提交一下数据库文件

c++编程学习





微信客服



QQ客服

 QQ客服

 kefu@csdn.net

 客服论坛

 400-660-0108

工作时间 8:30-22:00

关于我们 招聘 广告服务 网站地图

 百度提供站内搜索 京ICP证19004658号

©1999-2019 北京创新乐知网络技术有限公司

网络110报警服务 经营性网站备案信息

北京互联网违法和不良信息举报中心

中国互联网举报中心

1

码书商店



谷歌  
开源  
移动  
UI  
框架  
Flutter  
学习  
必读

出版社  
直供

关闭