C++语言程序设计 期中考试试题

考试时间：2012年4月17日

姓名： 学号： 班级：

**一、选择题（每小题1分，共40分）**

下列各题A）、B）、C）、D）四个选项中，只有一个选项是正确的。

1. 关于友元，下列说法错误的是（ ）

A)如果类A是类B的友元，那么类B也是类A的友元

B)如果函数fun()被说明为类A的友元，那么在fun()中可以访问类A的私有成员

C)友元函数可以减少系统开销，提高效率。

D)如果类A是类B的友元，那么类A的所有成员函数都是类B的友元

A

1. 下面选项中不属于面向对象程序设计特征的是（ ）。  
   A）继承性B）多态性C）类比性D）封装性

C

1. 已知下列语句中的x和y都是int型变量，其中错误的语句（ ）

A ) x=y++; B ) x=++y; C ) x++=y; D ) ++x=y;

C

1. 为了提高函数调用的实际运行速度，可以将较简单的函数定义为（　）。

A)内联函数 B)重载函数 C)递归函数 D)函数模板

A

1. 下列关于C++函数的叙述中，正确的是（ ）。  
   A）每个函数至少要具有一个参数   
   B）每个函数都必须返回一个值  
   C）函数在被调用之前必须先声明   
   D）函数不能自己调用自己

C

1. 有以下程序

#include<iostream>

int i = 0;

void fun(){

{ static int i = 1;

std::cout<<i++<<',';

}

std::cout<<i<<',';

}

int main(){

fun(); fun();

return 0;

}

程序执行后的输出结果是（ ）

1. 1,2,1,2, B) 1,2,2,3, C) 2,0,3,0, D) 1,0,2,0,

D

1. 下列字符串中，可以用作C++标识符的是  
   A）y\_2012 B）int C）3d D）foo-1

A

1. cout << 2^-5 << endl;的输出结果为（ ）

A）7 B）-5 C）0.03125 D）-7

D

1. 下列关于类和对象的叙述中，错误的是（ ）。  
   A）一个类只能有一个对象   
   B）对象是类的具体实例   
   C）类是对某一类对象的抽象   
   D）类和对象的关系是一种数据类型与变量的关系

A

10．设int a=3,b=2;，则++a\*b++的结果是（ ）。

A）6 B）8 C）9 D）12

B

11．在32位计算机中，一个int型数据所占用的内存长度为（ ）字节。

A）2 B）4 C）6 D）8

B

12．下列十六进制整型常量的写法中，错误的是（ ）。

A）0xaf B）0 C）0X1b D）0xAE

B

13. 下列变量定义中，正确的是（ ）

A）int m,n,x,y; float x,z; B）char c1,c2=”c”;

C）double age,do; D）float f1,\_time=0; double score\_;

D

14. 若有下面的函数调用：fun(a+b+c+d,3,min(a,b),n)

则fun函数的实参个数是（ ）。

A**．**3 B**．**4 C**．**5 D**．**6

B

15. 有如下函数定义：

void func(int a, int& b) {a++; b++;}

若执行代码段：

int x=2,y=2;

func(y,x);

则变量x和y的值分别是（ ）。

A**．**2和2 B**．**3和3 C**．**3和2 D**．**2和3

C

16. 在语句cout<<data中，cout是个（ ）

A）类　　B）对象　　C）C++关键字　　D）函数

B

17. 下列叙述中，错误的是

　　A）false是一个逻辑型常量　　 B）"b"是一个字符型常量

C）365是一个int常量　　 D）3.1415926是一个double常量

B

18. 执行语句序列

int n;

cin >> n;

switch(n)

{

case 1: cout << '1';

case 2: cout << '2';

case 3:

case 4: cout << '3'; break;

default: cout << '4';

}

时，若键盘输入1，则屏幕显示

A）1 B）14 C）12 D)123

D

19. 有如下程序：

int x = 4;

do{

x-=3;

cout << x;

}while(!(--x));

执行这个程序的输出结果是（ ）。

A．1 B．41 C．1-2 D．死循环

C

20. 已知枚举类型定义语句为： enum Token { NUMBER ,NAME, PLUS = 5, MINUS, PRINT = 10 };

则下列叙述中错误的是（ ）

A）枚举常量NAME的值为1

　 B）枚举常量NUMBER的值为1

　 C）枚举常量MINUS的值为6

　 D）枚举常量PRINT的值为10

B

21. 执行语句序列：int i=0; while(i < 25) i+=3; cout<<i; ，输出结果是（ ）。

　 A）24 　 B）25 　 C）27 　 D）28

C

22. 有如下程序：

#include<iostream>

using namespace std;

void fun (int x){

cout << x << endl;

}

int main()

{

cout << fun (23);

return 0;

}

如上程序的输出为（ ）

A）23 B）2323 C）编译错误 D）运行错误

C

23. 有如下程序：

#include <iostream>

using namespace std;  
void swap(int& x, int y) {

int t = x; x = y; y = t;

}  
int main( )  
{

int a=23,b=42;  
swap(b, a);  
cout << a << ", " << b<< endl;  
return 0;

}  
 执行后的输出结果是（ ）。  
　 A）42, 42 　 B）23, 23 　 C）23, 42 　 D）42, 23

B

24. 已知程序中已经定义了函数test，其原型是int test(int, int, int);，则下列重载形式中正确的是

A ) char test(int,int,int);

B ) double test(int,int,double);

C ) int test(int,int,int=0);

D ) float test(int,int,float=3.5F);

B

25. 有如下程序段：  
　　int i=1;  
　　while (1) {  
　　    i++;  
　　    if(i == 10) break;  
　    　if(i%2 == 0) cout << '\*';  
　　}  
　　执行这个程序段输出字符\*的个数是  
　　A）10　　 B）3　　 C）4　　 D）5

C

26. 计算斐波那契数列第n项的函数定义如下：  
　　int fib(int n){  
　　    if (n == 0) return 1;  
　　    else if (n == 1) return 2;  
　    　else return fib(n-1)+fib(n-2);  
　　}  
　　若执行函数调用表达式fib(2)，函数fib被调用的次数是  
　　A）1　　 B）2　　 C）3　　 D）4

C  
27. 有如下程序：

#include<iostream>

using namespace std;

int main()

{

void function(double val);

double val;

function(val);

cout<<val;

return 0;

}

void function(double val){

val = 3;

}

编译运行这个程序将出现的情况是

A）编译出错，无法运行 B）输出：3

C）输出：3.0 D）输出一个不确定的数

D

28. 有如下类声明：class Foo { int bar(); };  
　　则Foo类的成员bar是（ ）。  
　 A）公有数据成员 　 B）公有成员函数   
　 C）私有数据成员 　 D）私有成员函数

D

29. 若有语句 “int i=1,j=2; cout<<j/I+i/j;”, 则输出结果是（ ）

A)0 B)2.5 C)0.5 D)2

D

30. 在类声明中，紧跟在“public:”后声明的成员的访问权限是（ ）。

A）私有 B）公有 C）保护 D）默认

B

31. 关于函数重载，下列叙述中错误的是

A 重载函数的函数名必须相同

B 重载函数必须在函数形参个数或类型上有所不同

C 重载函数的返回值类型必须相同

D 重载函数的函数体可以有所不同

C

32. 以下关键字不能用来声明类的访问权限的是\_\_\_\_\_\_。  
　　A）public B）static C）protected D）private

B

33. 在下列函数原型中，可以作为类AA构造函数的是（ ）。  
　 A）void AA(int); 　B）int AA( ); 　C）AA(int)const; 　D）AA(int);

D

34. 有如下程序：

#include <iostream>

using namespace std;

class AA{

int n;

public:

AA(int k):n(k){ }  
int get( ){ return n;}  
int get( )const{ return n+1;}

};

int main( )

{

AA a(5);  
const AA b(6);  
cout<<a.get( )<<b.get( );  
return 0;

}  
执行后的输出结果是（ ）。  
A）55 　B）57 　C）75 　D）77

B

35. 有如下类定义：

class AA

{

int a;

public:

int getRef()const{ return &a; } //①

int getValue()const{ return a; } //②

void set(int n)const{ a=n; } //③

friend void show(AA aa)const{ cout<<a; } //④

};

其中的四个函数定义中错误的是（ ）

A）①③ B）② C）③ D）①③④

D

36. 有如下程序：

#include <iostream>

using namespace std;

class MyClass{

public:

MyClass() { cout << ‘\*’; }

MyClass(MyClass& a) { cout << ‘#’; }

~MyClass() { cout << ‘@’; }

};

int main(){

MyClass a;

MyClass b(a);

return 0;

}

运行时的输出结果是（ ）。

A)\*@#@ B)#@\*@ C)\*#@@ D)#\*@@

C

37. 有如下程序：

#include<iostream>  
using namespace std;  
class test｛

private:

int a;

public：

test() {cout<<"constructor"<<endl;}  
test(int a) {cout<<a<<endl;｝  
test（const test& test）{

a= test.a;  
cout<<”copy constructor"<<endl;

｝

~test（）｛cout<<"destructor"<<endl；｝

｝;

int main( )  
{

test A(3);

return 0;  
｝  
运行时输出的结果是

A） 3   
B） constructor  
 destructor  
C） copy constructor  
 destructor   
D） 3  
 destructor

D

38．对于一个类定义，下列叙述中错误的是（ ）

A） 如果没有定义拷贝构造函数，编译器将生成一个拷贝构造函数

B） 如果没有定义缺省的构造函数，编译器将一定生成一个缺省的构造函数

C） 如果没有定义构造函数，编译器将生成一个缺省的构造函数和一个拷贝构造函数

D） 如果已经定义了构造函数和拷贝构造函数，编译器不会生成任何构造函数

B

39. 有如下类定义：

class Foo

{

public:

Foo(int v) : value(v) { } // ①

~Foo() { } // ②

private:

Foo() { } // ③

int value = 0; // ④

};

其中存在语法错误的行是

A）① B）② C）③ D）④

D

40. 有以下程序

#include<iostream>

using namespace std;

class MyClass

{

public:

MyClass(int n){number = n;}

MyClass(MyClass &other){ number=other.number;}

~MyClass(){}

private:

int number;

};

MyClass fun(MyClass p)

{

MyClass temp(p);

return temp;

}

int main()

{

MyClass obj1(10), obj2(0);

MyClass obj3(obj1);

obj2=fun(obj3);

return 0;

}

程序执行时，MyClass类的拷贝构造函数被调用的次数是

A)5 B)4 C)3 D)2

B

**二、填空题（共60分）**

下面1-8每题1分，9-20每题2分，21-26每题3分，27-28每题5分。

1. C++是面向 的程序设计语言。对象

2. 在C++语言中，032表示一个\_\_\_\_八\_\_\_\_进制数，而0x23表示一个\_\_\_十六\_\_\_\_\_进制数。

1. 高级语言屏蔽了机器的细节，提高了语言的\_\_\_\_\_\_\_\_。抽象层次

4. C++中只有两个逻辑常量：true和 \_\_\_\_\_\_。false

5. 已知一个函数的原型是：int fn(double a);

若要以5.27为实参调用该函数，应使用表达式 \_\_\_\_\_\_\_\_。fn(5.27)

6. 非成员函数应声明为类的\_\_\_\_\_\_\_\_函数才能访问这个类的private成员。友元

7. 若有定义语句”int x=20,y=20 z=10;”，则表达式x>z&&y==z的布尔值为\_\_\_\_\_\_\_\_。false

8. 在类的对象被创建的时候，\_\_\_\_\_\_函数会被系统自动调用。构造

1. -7的二进制补码表示（8位）为 \_ \_ \_ \_ \_ \_ \_ \_。11111001
2. 执行语句序列:

int a=1,b=1;

b=(a=3\*4,a=a\*5);

cout<<b<<endl;

输出结果为\_\_\_\_\_\_\_\_\_\_。60

1. 若有定义语句：int a=3，b=2，c=1;，则表达式a<b?a:b的值是 。2
2. 执行语句序列:

int x=10,&r=x;

r++;

cout<<x<<'-'<<r<<endl;

输出结果为\_\_\_\_\_\_\_\_\_\_。11-11

1. 下列函数的功能是判断字符串str是否对称，对称则返回true，否则返回false。请在横线处填上适当内容，实现该函数。  
   bool fun(char \*str)  
   {

　 int i=0, j=0;  
　 while (str[j]) \_\_\_\_\_\_\_\_ ;j++  
　 for(j--; i<j && str[i]==str[j]; i++,j--);  
　 return i \_\_\_\_ j; >=  
　 }

14．下面程序运行的结果是 。\*#\*#\*#\*#$

#include <iostream>

using namespace std;

void main()

{

int i;

for (i=1;i<=6;i++){

if(i%3) cout <<"\*";

else continue;

cout <<"#";

}

cout <<"$"<<endl;

}

15. 已知递归函数f的定义如下：  
　　int f (int n)  
　　{

　　if(n<＝2) return 1;

　　else return n\*f(n-2);

}

则函数调用语句f(6)的结果是 。24

16. 执行下列语句后，变量sum的值是\_\_\_\_.7

int sum =1;

for (int i=1;i<=3;i++)

for (int j=3;j>=i;j--)

sum++;

17. 下列程序的输出结果为  
　Object id=1  
　Object id=2  
　请将程序补充完整。  
　#include <iostream>  
　using namespace std;  
　class Point   
　{  
　public:   
　 Point(int xx=0, int yy=0) {X=xx; Y=yy; countP++; }   
　 ~Point( ) { countP--; }  
　 static void GetC( ) {cout<<" Object id="<<countP><<endl;}   
　private:   
　 int X,Y;  
　 static int countP;   
　};  
　\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ //静态数据成员的初始化

int Point::countP=1;  
　int main( )   
　{  
　 Point::GetC( );  
　 Point A(4,5);   
　 A.GetC( );   
　 return 0;  
　}

18. 如下类定义中包含了构造函数和拷贝构造函数的原型声明，请在横线处填写正确的内容。

class myClass{

int data;

public:

myClass(int value); //构造函数

myClass(const \_\_\_\_\_\_\_\_\_\_\_\_ anotherObject); //拷贝构造函数

}

myClass&

19. 若下面程序运行时输出结果为

1,A,10.1

2,B,3.5

请将程序补充完整。

#include <iostream>

using namespace std;

int main()

{

void test(int, char, double \_\_\_\_\_\_\_\_\_\_\_ );c=3.5

test(1, 'A', 10.1 );

test(2, 'B');

return 0;

}

void test(int a, char b, double c){

cout<<a<<','<<b<<','<<c<<endl;

}

20. 有如下类声明：

Class MyClass

{

int i;

private: int j;

public: int m,n;

};

其中，私有成员的数量为\_\_\_\_\_\_\_\_。2

21．写出执行完下列代码段之后指定变量的值：

bool x =false,y=true,z=false;

x=x&&y||z;false

y=x||y&&z;false

z= !(x!=y) || (y==z);true

则x= ; y= ; z= ;

22. 字面常量42, 4.2, 42L的数据类型分别是\_\_\_\_\_\_、\_\_\_\_\_\_、\_\_\_\_\_\_。int double long

23. 下面程序的运行结果为\_\_\_\_\_\_\_\_。43

#include <iostream>

class A{

int num;

public:

A(int i)(num=++i;}

A(A &a){num=a.num++;}

void print()(cout<<num;}

}

void main()

{

A a(2),b(a);

a.print();

b.print();

}

24. 下面程序段的运行结果为 ：a=-5

int i=0,a=0,s=1;

do{

a=a+s\*i;

s=-s;

i++;

}while(i<10);

cout <<"a="<<a<<endl;

25. 已知函数f的定义如下：

int f (int n){  
　　 cout<<n;

return ++n;

}

则函数调用语句f(f(2))的结果是 。23

26. 有4个由用户任意输入的整数a、b、c和d，要求按照从大到小的顺序输出，请补充以下程序使其满足要求。

#include <iostream>

using namespace std;

void main()

{

int a, b, c, d, t;

cin >>a>>b>>c>>d;

if (a<b) { t =a; a=b; b=t;}

if (a<c) { t =a; a=c; c=t;}

if (a<d) { t =a; a=d; d=t;}

if (\_\_\_\_\_){ t =b; b=c; c=t;} b<c

if (b<d) { t =b; b=d; d=t;}

if (c<d) { t =c; c=d; d=t;}

cout <<a<<","<<b<<","<<c<<","<<d<<endl;

}

27. 实现两个重载函数，使得下面程序的结果为：

3

4

#include <iostream>

using namespace std;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

int main(){

int a=2,b=3,c=4;

double x=5.5,y=2.1,z=4;

cout<<middle(a,b,c)<<endl;

cout<<middle(x,y,z)<<endl;

return 0;

}

28. 有如下程序:

#include <iostream>

using namespace std;

class Exam{

public:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

void print(){cout<<a<<b<<endl;}

private:

int a;

const int b;

};

int main(){

Exam obj(1,2);

obj.print();

return 0;

}

实现Exam类的构造函数和析构函数，使得程序的输出结果为：

开始构造

12

开始析构

**2012年秋季学期《计算机程序设计基础》期中试题（A卷）**

**答卷**

**学号\_\_\_\_\_\_\_\_\_ 班级\_\_\_\_\_\_\_\_\_ 姓名\_\_\_\_\_\_\_\_\_ 老师名\_\_\_\_\_\_\_\_\_ 得分\_\_\_\_\_\_\_\_\_**

**一、选择题**

1-5： \_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_

6-10： \_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_

11-15：\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_

16-20：\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_

21-25：\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_

26-30：\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_

31-35：\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_

36-40：\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_

**二、填空题（如果有多个填空，答案之间用空格或者顿号隔开）**

1：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

2：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

3：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

4：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

5：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

6：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

7：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

8：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

9：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

10：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

11：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

12：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

13：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

14：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

15：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

16：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

17：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

18：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

19：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

20：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

21：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

22：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

23：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

24：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

25：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

26：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**27.**

#include <iostream>

using namespace std;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

int main(){

int a=2,b=3,c=4;

double x=5.5,y=2.1,z=4;

cout<<middle(a,b,c)<<endl;

cout<<middle(x,y,z)<<endl;

return 0;

}

**28.**

#include <iostream>

using namespace std;

class Exam{

public:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

void print(){cout<<a<<b<<endl;}

private:

int a;

const int b;

};

int main(){

Exam obj(1,2);

obj.print();

return 0;

}