

全国 2015 年 4 月高等教育自学考试

C++ 程序设计试题

课程代码:04737

请考生按规定用笔将所有试题的答案涂、写在答题纸上。

选择题部分

注意事项:

1. 答题前,考生务必将自己的考试课程名称、姓名、准考证号用黑色字迹的签字笔或钢笔填写在答题纸规定的位置上。
2. 每小题选出答案后,用 2B 铅笔把答题纸上对应题目的答案标号涂黑。如需改动,用橡皮擦干净后,再选涂其他答案标号。不能答在试题卷上。

一、单项选择题(本大题共 20 小题,每小题 1 分,共 20 分)

在每小题列出的四个备选项中只有一个是符合题目要求的,请将其选出并将“答题纸”的相应代码涂黑。错涂、多涂或未涂均无分。

1. C++ 语言是从早期的 C 语言逐渐发展演变而来的,与 C 语言相比,C++ 在求解问题方法上进行的最大改进是
A. 面向过程 B. 面向对象 C. 安全性 D. 复用性
2. 结构化程序设计所规定的三种基本控制结构是
A. 输入、处理、输出 B. 树形、网形、环形
C. 顺序、选择、循环 D. 主程序、子程序、函数
3. 关于函数模板,描述错误的是
A. 函数模板必须由程序员实例化为可执行的函数模板
B. 函数模板的实例化由编译器实现
C. 一个类定义中,只要有一个函数模板,则这个类是类模板
D. 类模板的成员函数都是函数模板,类模板实例化后,成员函数也随之实例化
4. 作用域运算符“::”的功能是
A. 标识作用域的级别的 B. 指出作用域的范围的
C. 给定作用域的大小的 D. 标识成员是属于哪个类的
5. 采用重载函数的目的是
A. 实现共享 B. 减少空间
C. 提高速度 D. 使用方便,提高可读性
6. 以下关键字不能用来声明类的访问权限的是
A. public B. private C. static D. protected
7. 有如下类声明“class A {int x; ...};”,则 A 类的成员 x 是
A. 公有数据成员 B. 私有数据成员
C. 公有成员函数 D. 私有成员函数

8. 下列不是构造函数的特征的是
- A. 构造函数的函数名与类名相同
 - B. 构造函数可以重载
 - C. 构造函数可以设置缺省参数
 - D. 构造函数不能重载
9. 通常的拷贝构造函数的参数是
- A. 某个对象名
 - B. 某个对象成员名
 - C. 某个对象的引用
 - D. 某个对象的指针名
10. 下列哪个类型函数不适合声明为内联函数
- A. 函数体语句较多
 - B. 函数体语句较少
 - C. 函数执行时间较短
 - D. 函数执行时间过长
11. 设 x 和 y 均为 `bool` 量, 则 $x \& \& y$ 为真的条件是
- A. 它们均为真
 - B. 其中一个为真
 - C. 它们均为假
 - D. 其中一个为假
12. 对于友元描述正确的是
- A. 友元是本类的成员函数
 - B. 友元不是本类的成员函数
 - C. 友元不是函数
 - D. 友元不能访问本类私有成员
13. 当一个派生类公有继承一个基类时, 基类中的所有公有成员成为派生类的
- A. `public` 成员
 - B. `private` 成员
 - C. `protected` 成员
 - D. 友元
14. 在 C++ 语言中, 080 是
- A. 八进制数
 - B. 十进制数
 - C. 十六进制数
 - D. 非法数
15. 下面函数原型声明中, 声明了 `fun` 为虚函数的是
- A. `void fun() = 0`
 - B. `virtual void fun() = 0`
 - C. `virtual void fun()`
 - D. `virtual void fun() {}`
16. 下列虚基类的声明中, 正确的是
- A. `class virtual B: public A`
 - B. `class B: virtual public A`
 - C. `class B: public A virtual`
 - D. `virtual class B: public A`
17. 下列运算符中, 在 C++ 语言中不能重载的是
- A. `*`
 - B. `>=`
 - C. `::`
 - D. `/`
18. 在 C++ 语言中, 能代表逻辑值“假”的是
- A. 小于 0 的数
 - B. 大于 0 的数
 - C. 非 0 整数
 - D. 0
19. 以下叙述中不正确的是
- A. 在一个函数中, 可以有多条 `return` 语句
 - B. 函数的定义不能嵌套, 但函数的调用可以嵌套
 - C. 函数必须有返回值
 - D. 不同的函数中可以使用相同名字的变量
20. 已知: `int m = 10;` 在下列表示引用的方法中, 下列哪一项是正确的
- A. `int &x = m`
 - B. `int &y = 10`
 - C. `int &z`
 - D. `float &t = &m`

非选择题部分

注意事项:

用黑色字迹的签字笔或钢笔将答案写在答题纸上,不能答在试题卷上。

二、填空题(本大题共 20 小题,每小题 1 分,共 20 分)

21. 对象的核心概念就是通常所说的继承、封装和_____。
22. cin 后面的符号是: >>, cout 后面的符号是:_____。
23. 面向对象编程的英文缩写是:_____。
24. 已知 $x=2, y=3, x < y$ 的值是:_____。
25. 执行下列代码:
`double pi=3.1415926; cout << setprecision(5) << pi;`
程序的输出结果是_____。
26. 建立一个对象时,对象的状态是不确定的。为了使对象的状态确定,必须对其进行正确的_____。
27. 重载运算符的含义必须清楚,不能有_____。
28. 按照联编进行阶段的不同,联编方法可分为两种:动态联编和_____。
29. 设要把一个文件输出流对象 myFile 与文件“f:\myText.txt”相关联,所用的 C++ 语句是:_____。
30. 在 C++ 中,函数的参数有两种传递方式,它们是_____和地址或指针或引用传递。
31. 设有定义 `int x=3, y=4`; 则表达式 `x*=y+5` 的值为_____。
32. 在 C++ 中,函数在定义时可以预先定义出默认的形参值。默认形参值必须按_____的顺序定义。因为在调用时,实参初始化形参是按从左到右的顺序。
33. 如果一个函数直接或间接地调用自身,这样的调用称为_____调用。
34. 函数模板中紧随 template 之后尖括号内的类型参数都要冠以保留字_____。
35. 假如一个类的名称为 MyClass,使用这个类的一个对象初始化该类的另一个对象时,可以调用_____构造函数来完成此功能。
36. 设函数 sum 是由函数模板实现的,并且 `sum(3,6)` 和 `sum(4.6,8)` 都是正确的函数调用,则函数模板具有_____个类型参数。
37. 如果一个派生类只有一个唯一的基类,则这样的继承关系称为_____。
38. 一个 const 对象只能访问_____成员函数。
39. 如果通过同一个基类派生一系列的类,则将这些类总称为_____。
40. A 是一个类,已有语句“`A * p; p=new A[10];`”。要释放由 p 指向的动态空间,正确的语句应该是_____。

三、改错题(本大题共 5 小题,每小题 4 分,共 20 分)

下面的程序段中有一处错误,请用下划线标出错误所在行并指出错误原因

41.

```
#include "stdafx.h"
#include <iostream>
using namespace std;
class Test
{ private:
int x,y = 20;
public:
Test(int i,int j) { x = i,y = j; }
int getx() { return x; }
int gety() { return y; }
};
void main()
{ Test mt(10,20);
cout << mt.getx() << endl;
cout << mt.gety() << endl;
}
```

www.zikao365.com

42.

```
#include "stdafx.h"
#include <iostream>
using namespace std;
class Test
{ int x,y;
public:
fun(int i,int j)
{ x = i;y = j; }
show()
{ cout << "x = " << x;
if(y)
cout << ", y = " << y << endl;
cout << endl; }
};
void main()
{ Test A;
A.fun(1);
A.show();
A.fun(2,4);
A.show();
}
```

43.

```
#include "stdafx.h"
#include <iostream>
using namespace std;
class A
{ int i;
public:
virtual void fun() = 0;
A( int a)
{ i = a; }
};
class B:public A
{ int j;
public:
void fun()
{ cout << "B::fun()\n"; }
B( int m, int n = 0) : A(m), j(n) {}
};
void main()
{ A *pa;
B b(7);
pa = &b;
}
```

44.

```
#include <iostream.h>
class f{
private: T x,y;
public: f1(T a,T b){ x = a;y = b; }
float max() { return(x > y)? x:y; }
};
main() {
f a;
a.f1(1.5,3.8);
cout << a.max() << endl;
}
```

45.

```
#include <iostream.h>
#include <string.h>
class Base
{ public:
Base(char *s = "\0") { strcpy(name,s); }
void show();
protected:
char name[20];
};
```

```

Base b;
void show()
{ cout << "name:" << b.name << endl; }
void main()
{ Base d2("hello");
show();
}

```

四、完成程序题(本大题共 5 小题,每小题 4 分,共 20 分)

46. 在下面程序横线处填上适当字句,以使该程序执行结果为:

50 4 34 21 10

6.1 7.1 8.1 9.1 10.1 11.1

```
#include <iostream>
```

```
using namespace std;
```

```
template <class T>
```

```
void f ( _____ )
```

```
{ _____;
```

```
for ( int i=0;i<n/2;i++ )
```

```
t = a[i], a[i] = a[n-1-i], a[n-1-i] = t;
```

```
}
```

```
void main ( )
```

```
{ int a[5] = {10,21,34,4,50};
```

```
double d[6] = {11.1,10.1,9.1,8.1,7.1,6.1};
```

```
f(a,5);f(d,6);
```

```
for ( int i=0;i<5;i++ )
```

```
cout << a[i] << " ";
```

```
cout << endl;
```

```
for ( int i=0;i<6;i++ )
```

```
cout << d[i] << " ";
```

```
cout << endl;
```

```
}
```

47. 在下面程序的底画线处填上适当的字句,使该程序执行结果为 40,x 为静态成员。

```
#include <iostream.h>
```

```
class Test
```

```
{ public:
```

```
_____;
```

```
Test ( int i=0)
```

```
{ x = i + x; }
```

```
int Getnum()
```

```
{ return Test::x + 10; }
```

```
};
```

```
_____;
```

```
void main()
```

```
{ Test test;
```

```
cout << test.Getnum() << endl;
```

```
}
```

48. 在下列程序的空格处填上适当的字句,使输出为:0,2,10。

```
#include "stdafx. h"
#include <iostream>
#include "math. h"
using namespace std;
class Magic
{ double x;
public:
Magic( double d = 0. 00 ) : x( fabs( d ) )
{ }
Magic operator + ( _____ )
{
return Magic( sqrt( x * x + c. x * c. x ) );
}
_____ << ( ostream & stream, Magic & c )
{ stream << c. x;
return stream;
}
};
void main( )
{ Magic ma;
cout << ma << " , " << Magic( 2 ) << " , " << ma + Magic( - 6 ) +
Magic( - 8 ) << endl;
}
```

www.zikao365.com

49. 下面是一个输入半径,输出其面积和周长的 C++ 程序,在下划线处填上正确的语句。

```
#include "stdafx. h"
#include <iostream>
#include "math. h"
_____
_____
void main( )
{ double rad;
cout << "rad = " ;
cin >> rad;
double l = 2. 0 * pi * rad;
double s = pi * rad * rad;
cout << " \n The long is:" << l << endl;
cout << "The area is:" << s << endl; }
```

50. 请完成一下程序,使其输出结果为:

```
x = 5, y = 27
x = 10, y = 27

#include <iostream>
using namespace std;
class Sample
{ private:
int x;
static int y;
_____ :
Sample( int a );
void print( ) ;
};
```

```

Sample::Sample(int a)
{
    x = a;
    y++;
}
void Sample::print()
{
    cout << "x = " << x << ", y = " << y << endl;
}

_____;
void main()
{
    Sample s1(5);
    Sample s2(10);
    s1.print();
    s2.print();
}

```

五、程序分析题(本大题共 2 小题,每小题 5 分,共 10 分)

51. 写出程序运行结果

```

#include "stdafx.h"
#include <iostream>
using namespace std;
class B; class A
{ public:
    A(int i) { a = i; }
    friend int F(A &f1, B &f2);
private:
    int a;
};
class B
{ public:
    B(int i) { b = i; }
    friend int F(A &f1, B &f2);
private:
    int b;
};
int F(A &f1, B &f2)
{ return (f1.a + f2.b) * (f1.a - f2.b); }
void main()
{
    A n1(10);
    B n2(8);
    cout << F(n1, n2) << endl; }

```

52. 写出程序运行结果

```

#include <iostream.h>
int func(int a)
{ int b = 0;
    static int c = 4;
    b++; c--;
    return(a + b + c); }

```



```

void main( )
{ int a =2;
for( int j =0;j <2;j++ )
cout << func( a + j) << " ";
}

```

六、程序设计题(本大题共 1 小题,每小题 10 分,共 10 分)

53. 声明复数类,Complex,该类中有两个私有变量 real, image 分别表示一个复数的实部和虚部。为 Complex 类添加适当的构造函数。并使用友元函数 add 实现复数加法。

```

#include <iostream>
using namespace std;
class Complex
{
private:
double real, image;
public:
void setRI( double a, double b)
{
real = a;
image = b;
}
double getReal()
{
return real;
}
double getImage()
{
return image;
}
void print() {
if( image > 0)
cout << "复数:" << real << " + " << image << "i" << endl;
if( image < 0)
cout << "复数:" << real << " - " << image << "i" << endl;
}
friend Complex add( Complex ,Complex );//声明友元函数
};

void main()
{
Complex c1(19, 0.864), c2, c3;
c2.setRI(90,125.012);
c3 = add(c1, c2);
cout << "复数一:"; c1.print();
cout << "复数二:"; c2.print();
cout << "相加后:"; c3.print();
}

```