



北京科技大学 2016-2017 学年 第二学期

程序设计基础 (C++) 期末模拟试卷

一、选择题

1. 求 “abc\\12\\n” 字符串的长度 ()
(A) 5 (B) 6 (C) 7 (D) 8
2. 若变量都已正确定义并赋值, 下面合法的表达式是 ()。
 (float) a=b+7 (B) a=7+b+c, ++a
 int (12.4/3) (D) a=a+2=c+b
3. 语句 while(w) ... 中的表达式 w 的等价表示是: ()
(A) w==0 (B) w==1 (C) w!=0 (D) w!=1
4. 语句 while(a>b) a--; 等价于 ()。
(A) if(a>b) a--; (B) do {a--} while(a>b);
(C) for(a>b) a--; (D) for(;a>b; a--);
5. 设 t 是 double 类型变量, 表达式 t=1, t+2, t++ 的值是 ()
(A) 4.0 (B) 3.0 (C) 2.0 (D) 1.0
6. 执行以下程序, 输出结果为 ()

```
#include<stdio.h>
```

```
Void main()
```

```
{
```

```
int a=10,b=0;
```

```
if(a==10)
```

```
    a+=1;b+=1;
```

```
else
```

```
    a+=4;b+=4;
```

```
cout<<a<<" , " <<b;
```

}

(A) 11, 1

(B) 14, 1

(C) 14, 4

(D) 有语法错误

7. 若二维数组 y 有 m 列, 则位于 $y[i][j]$ 之前的元素个数有 () 个

(A) $j*m+i$

(B) $i*m+j$

(C) $i*m+j-1$

(D) $i*m+j+1$

8. 下列数组的定义中, () 是错误的。

(A) `char ca1[]={ 'c' , ' h' , ' a' , ' r' };` (B) `char ca2[5]=" char" ;`

(C) `char ca3[4]=" char"`

(D) `int array[]={6, 5, 3, 4};`

9. 下面关于 C++ 字符数组的叙述中, 错误的是 ()。

(A) 字符数组可以放字符串

(B) 字符数组中的字符串可以进行整体输入、输出

(C) 可以在赋值语句中通过赋值运算符 “=” 对字符数组整体赋值

(D) 字符数组的下标从 0 开始

10. 下面判断是否构成重载函数的条件中, 错误的判断条件是 ()

(A) 参数类型不同 (B) 参数个数不同 (C) 参数顺序不同 (D) 函数返回值不同

11. 假设 a, b 是全局变量, 下面设置函数参数默认值的函数原型中, 错误的是

()

(A) `int fun(int x, int y=10);`

(B) `int fun(int x=5, int y=10);`

(C) `int fun(int x=5, int y);`

(D) `int fun(int x, int y=a+b);`

12. C++ 中函数返回值的类型是由 () 决定的。

(A) 函数定义时指定的类型

(B) `return` 语句中的表达式类型

(C) 调用该函数时的实参的数据类型

(D) 形参的数据类型

13. 以下正确的函数原型声明语句是 ()

(A) `int fun(int a, b);`

(B) `float fun(int a; int b);`

(C) double fun();

(D) int fun(char a[][]);

14. 已有定义 “int a=5,&ra=a;” 则下叙述中, 错误的说法是 ()。

(A) ra 是变量 a 的引用 (B) ra 的值为 5

(C) ra 是 a 的地址值 (D) 执行 “ra=10;” 后变量 a 的值也变为 10

15. 设有以下定义和语句:

```
int a[3][2]={1, 2, 3, 4, 5, 6}, *p=[3];
```

```
p[0]=a[0];
```

则*(p[0]+1) 所代表的数组元素是 ()

(A) a[0][1]

(B) a[1][0]

(C) a[1][1]

(D) a[1][2]

16. 下面关于 new 和 delete 运算的叙述中, 错误的说法是 ()。

(A) 由 new 分配的内存空间是连续的

(B) 由 new 运算符分配的内存空间, 当函数执行结束时系统会自动收回

(C) 如果当前内存无足够的空间可分配, 则 new 运算符返回 NULL

(D) 对于程序中的静态数组占用的存储空间不能使用 delete 来释放

17. 当定义一个联合体变量时系统分配给它的内存字节数是 ()

(A) 联合体中第一个成员所需的内存字节数

(B) 联合体中最后一个成员所需的内存字节数

(C) 联合体中占用内存最大的成员所需的字节数

(D) 联合体中各成员所需的内存字节数和

18. 关于类的描述, 不正确的是 ()

(A) 类是一种数据类型

(B) class 默认访问权限是 private

(C) 类可以没有任何构造函数

(D) 对象是类的实例化



```
class Person{ Person() {cout << 1;} ~Person() {cout << 2;}}
```

```
void f() {Person p2, *p = new Person(); }
```

函数 f 调用完成后的输出结果是()

- (A) 1122 (B) 11 (C) 122 (D) 112

20. 关于友元的描述, 不正确的是()

- (A) 一个类可以是另一个类的友元
(B) 一个类的某个成员函数可以是另一个类的友元
(C) 友元加强了类的封装性
(D) 一个类的友元函数可以访问其私有变量

21. 构造函数重载的作用是()

- (A) 实现数据的隐藏 (B) 方便使用, 提高编程效率
(C) 降低代码重用性 (D) 提高编译和运行的速度

```
22. class Father{public: Father(int n){ cout << n << "Father" ;}}  
class Son: Father{public: Son(int n){ cout << n << "Son" ;}}
```

则语句 Son a(3); 的输出结果是()

- A. 3Father B. 3Son C. 3Father3Son D. 编译错误, 无法执行

23. 关于基类和派生类的关系, 下列正确的是()

- (A) 派生类指针可以指向基类对象
(B) 基类的引用不可以引用派生类对象
(C) 基类对象可以赋值给派生类对象
(D) 函数形参是基类对象, 则实参可以是派生类对象。





 关于抽象基类的描述, 不正确的是()

- (A) 抽象基类不能被实例化
(B) 有纯虚函数的类是抽象基类
(C) 派生类必须重写抽象基类的所有纯虚函数才能实例化
(D) 抽象基类不能有非纯虚函数

25. 关于多态的描述, 下列正确的是()

- (A) 多态是编译期间发生的
(B) 派生类调用基类的成员函数就会发生多态
(C) 只有使用基类的指针或引用时才可能发生多态
(D) 基类调用派生类的成员函数就会发生多态




二、填空题


1. 一个简单的 C++ 程序由注释、 和  组成。
2. 使用 C++ 语言开发程序的过程包括：编辑、、 和运行。
3. 已知 `int x=1, y=0;` 执行下面程序段后 `if(x) { if (x>0) y=1; } else y = -1;` 后 `y` 的值为_____。
4. 定义下列变量和数组：

```
int k;
```

```
int a[3][3]={9, 8, 7, 6, 5, 4, 3, 2, 1};
```

则语句 `for (k=0;k<3;k++) cout<<a[k][k];` 的输出结果是_____。

5. 函数 `fun(x, n)` 的功能是计算 x^n 。main 函数中已经正确定义 `m, a, b` 变量并赋值，实现计算 $m=a^4+b^4-(a+b)^4$ 正确的调用语句为：_____。
6. C++ 中形参的默认存储类别是 。
7. 现有一个 `Person` 类，则语句 `Person a, *b = &a, c[100];` 执行后，共调用了  次 `Person` 类的构造函数。
8. 销毁派生类对象时，先调用  的析构函数。
9. 派生类只能访问基类的_____、_____成员（填访问权限关键字）。

```
 class Base{ virtual void f() { cout << "Base" ;}}
```

```
class AA: Base{ void f() { cout << "AA" ;}}
```

```
AA a; Base b = a, &c = a, *p = &c;
```

则 `b.f()`、`c.f()`、`p->f()` 的输出结果

分别是_____、_____、_____。

三、读程序写结果

1. 输入为 5 9

```
#include<iostream>
using namespace std;
int main()
{   int a,b;
    int *p1,*p2,*p3;
    cin>>a>>b;
    p1=&a, p2=&b;
    if(a<b)
    {
        p3=p1;
        p1=p2;
        p2=p3;
    }
    cout<<*p1<<endl
    <<*p2<<endl;
    return 0;}
```



北京科技大学学生学习与发展指导中心
Center for Student Learning and Development USTB



```
#include<iostream>
using namespace std;
class Person{
public:
    char *name;
    int age;
    Person(int, char*);
    Person();
    ~Person();
```

```

        Person(const Person &anotherPerson);
};

Person:: Person() {
    cout << "person" << endl;
}


Person:: Person(const Person &anotherPerson) {
    name = anotherPerson.name;
    age = anotherPerson.age;
    cout << "copied" << endl;
}

Person:: Person(int pAge, char *pName) {
    age = pAge;
    name = pName;
}

Person::~ ~Person() {
    cout << name << "destroyed!" << endl;
}

int main() {
    char bname[10] = "bname";
    Person b(3, bname);
    Person a(b);
    Person *c = new Person();
    c->name = "abc";
    strcpy(bname, "newname");
    return 0; }

```

程序开始运行至结束，输出的信息分别是 、____、____、____。

3.

```

#include<iostream>

using namespace std;

```

```
class Father {
public:
    int age;
    Father();
    Father(int);
    ~Father();
};

Father::Father() {
    cout << "Father" << endl;
}

Father::Father(int a) {
    age = a;
    cout << "Father" << a << endl;
}

Father::~~Father() {
    cout << "Father Destroyed" << endl;
}

class Son : Father {
public:
    int height;
    Son(int age, int height);
    Son();
};

Son::Son() {
    cout << "Son" << endl;
}

Son::Son(int age, int height) : Father(age) {
    cout << "Son" << height << endl;
}
```

```
int main() {  
    Son s1, s2(15, 165);  
    return 0; }
```

程序开始运行至结束，输出的信息

分别是_____、_____、_____、_____、_____、_____。

四、程序填空

1. 利用指针，编写用于交换两个整型变量值的函数。程序运行结果如下：

输入：5 6

输出：6 5

```
#include <iostream>  
using namespace std;  
void swap(int *xp, int *yp)  
{  
    int tmp;  
    _____ (1) _____  
    *xp=*yp;  
    _____ (2) _____  
}  
  
int main()  
{  
    int a,b;  
    cin>>a>>b;  
    swap(&a, _____ (3) _____);  
    cout<<a<<" "<<b<<endl;  
    return 0;  
}
```

北京科技大学学生学习与发展指导中心
Center for Student Learning and Development USTB

2. 编写主程序，将输入字符串反序输出。程序运行结果如下：

输入: ABCDEFGHIJK

输出: KJIHGFEDCBA

```
#include<iostream>
#include<cstring>
using namespace std;
int main()
{
    char str[100];
    cin>>str;
    _____ (1) _____
    len=strlen(str);
    char *p=&str[len-1];
    while(p>=str)
    {
        cout<<*p;
        _____ (2) _____
    }
```

```
        cout<<endl;
```

```
        return 0;
```

北京科技大学学生学习与发展指导中心
Center for Student Learning and Development USTB

3. 程序输出结果是分别是 50 和 11, 填空

```
#include<iostream>
using namespace std;
class Father {
public:
    static int count;
    int age;
    Father();
    ~Father();
```

```

};

_____ (1) _____=0;
Father::Father() {
    age = 50;
    ++count;
}
Father::~~Father() {
    --count;
}

class Son : public Father {
public:
    int age;
    Son(int);
};
Son::Son(int age) {
    this->age = age;
}
void f() {
    Father f1, f2[100], *f3 = new Father[10];
}
int main() {
    f();
    Son s(18);
    cout << __ (2) __age<< endl;
    cout << __ (3) __ count << endl;
    return 0;
}

```

4. 程序的输出结果分别是 Plant、Vegetable、Cucumber10、Plant。填空

```
#include<iostream>

using namespace std;

class Plant {
public:
    int size;
    virtual void f () {
        cout << "Plant" << endl;
    }
};

class Fruit: public virtual Plant {
public:
    void print() {
        cout << "Fruit"<< endl;
    }
};

class Vegetable : public ____ (1) ____ {
    void print() {
        cout << "Vegetable" << endl;
    }
};

class Cucumber : public Fruit, public Vegetable {
    void print() {
        cout << "Cucumber" << size << endl;
    }
}

int main()
{
    Plant *p1 = new Fruit();
    Plant *p2 = new Vegetable();
```

```
Plant_____ (2) _____ = *(new Cucumber());  
p3.size = 10;  
Plant p4 = *p1;  
p1->_____ (3) _____;  
p2->print();  
p3.print();  
p4.print();  
return 0;  
}
```



北京科技大学学生学习与发展指导中心
Center for Student Learning and Development USTB