

北京邮电大学 2020——2021 学年第一学期

《C/C++程序设计与编程方法》期末考试试题（A）

考试 注意 事项	一、学生参加考试须带学生证或学院证明，未带者不准进入考场。学生必须按照监考教师指定座位就坐。 二、书本、参考资料、书包等物品一律放到考场指定位置。 三、学生不得另行携带、使用稿纸，要遵守《北京邮电大学考场规则》，有考场违纪或作弊行为者，按相应规定严肃处理。 四、学生必须将答题内容做在试题答卷上，做在试题及草稿纸上一律无效。 五、学生的姓名、班级、学号、班内序号等信息由教材中心统一印制。								
考试 课程				考试时间		年 月 日			
题号	一	二	三	四	五	六	七	八	总分
满分	23	30	15	12	20				
得分									
阅卷 教师									

一. 填空题：(每空 1 分，共 23 分)

1、C 或 C++源程序经过编译生成_____文件，再通过_____生成可执行文件。

2、C 或 C++字符串“xy\\a”，其中有_____个有效字符，在机器中存储占_____字节。

3、若定义 `int x=0xFFFFFBA;` 则执行表达式 `x&=_____` 后，x 的值变为 `0xFF0000BA`。

4、程序段

```
for(int i=0;i<100;i+=3){cout << i << endl;}
```

共输出显示_____行。

5、以下程序段代码的输出结果是_____和_____。

```
char s[]="abcdefg";
```

```
s[3]=s[5]=0;
```

```
cout<<s<<endl;
```

```
cout<<s+4<<endl;
```

- 6、数组 `unsigned short int a[3][5]`一共有_____个元素。该数组占_____字节。
- 7、枚举类型定义为：`enum COLOR{ WHITE, BLACK=100, RED, BLUE, GREEN=300}`; 则 `WHITE` 和 `BLUE` 的值分别为_____和_____。
- 8、若定义变量 `int x=0, y=10`, 若在执行函数调用语句 `swap(x, y);` 后, `x` 和 `y` 值发生了交换, 则声明无返回类型 `swap` 函数的原型应为: _____。
若在执行函数调用语句 `swap(&x, &y);` 后, `x` 和 `y` 值发生了交换, 则声明无返回类型 `swap` 函数的原型应为: _____。
- 9、若定义指针数组 `const char *a[]={ "Beijing", "Shanghai" }`; 则执行 `cout << a[0];` 输出为_____, 执行 `cout << a[0][1];` 输出为_____, 执行 `cout << a[1][0];` 输出为_____。
- 10、每当创建某个类的对象都会自动调用其_____函数, 而对象生存期结束时会自动调用类的_____函数。
- 11、类的数据成员和成员函数的访问控制属性有: `public`、_____和_____。其中_____属性的成员可以被类的对象直接访问。若某成员不写访问控制属性关键字, 则其默认是_____。

二. 单项选择题 (每题 1 分, 共 30 分)

1. 关于 C++ 语言和 C 语言的关系, 下列叙述不正确的是 ()
- A. C++ 语言是从 C 语言发展演变而来的
 - B. C++ 语言是对 C 语言进行了改进和扩充
 - C. C++ 语言全面兼容 C 语言, 并支持面向对象的程序设计
 - D. C++ 语言和 C 语言没有任何关系, 是全新开发的一门编程语言
2. 在面向对象程序设计中, 区别于结构化程序设计的基本特征是 ()
- A. 自顶向下, 逐步求精
 - B. 模块化
 - C. 顺序、选择、循环 3 种基本结构
 - D. 封装性
3. 下列变量名不合法的是 ()
- A. `Break`
 - B. `_123`
 - C. `"123"`
 - D. `M123`

4. a 为整型变量，下列 C++表达式中不能正确表达数学关系 $5 > a > 1$ 的是 ()

- A. $a < 5 \ \&\& \ a > 1$
- B. $!(a \leq 1 \ \parallel \ a \geq 5)$
- C. $1 < a < 5$
- D. $a == 2 \parallel a == 3 \parallel a == 4$

5. 以下语句输出结果为 4 的是 ()

- A. `cout<<5/10*8<<endl;`
- B. `cout<<5/10*8.0<<endl;`
- C. `cout<<5/10.0*8<<endl;`
- D. `cout<<double(5/10*8)<<endl;`

6. 使用标准库的 ifstream 类来定义文件流对象，进行文件的读操作时，应包含头文件 ()

- A. iostream
- B. fstream
- C. iomanip
- D. string

7. 定义 `int a=1;` 下列循环体内的语句执行次数是 ()

```
do
{
    a--;
}while(!a);
```

- A. 0
- B. 1
- C. 2
- D. 无限

8. 下面程序运行结果是 ()

```
int i=1, sum=0;
while(i<=15)
    if(i++%3!=2)
        continue;
    else
        sum+=i;
cout<<"sum="<<sum<<endl;
```

- A. sum=40
- B. sum=45
- C. sum=50
- D. sum=55

9. 数组 A 和数组 B 都是具有 10 个元素的 int 型数组，下面能够正确实现把数组 A 复制到数组 B 的 C++代码是 ()

- A. B = A;
- B. strcpy(B, A);
- C. B[10] = A[10];
- D. for(int i=0;i<10;B[i]=A[i],i++) ;

10. 枚举类型定义为 enum Shape{TRIANGLE, CIRCLE, RECTANGLE}; 下列语句中正确的是 ()

- A. Shape s = CIRCLE;
- B. CIRCLE c;
- C. CIRCLE = 5;
- D. cout<<Shape.CIRCLE;

11. 若有以下定义，则下列赋值不正确的是 ()

```
struct Person
{
    char name[50];
    double salary;
}person[2]={ {"Li",5000} };
```

- A. person[1].name = person[0].name;
- B. person[1].salary = person[0].salary;
- C. person[1].name[0] = person[0].name[0];
- D. person[1] = person[0];

12. 若有以下定义，则语句 cout<<sizeof(p); 的执行结果是 ()

```
union udata
{
    char pos[20];
    int num[20];
} *p;
```

- A. 4
- B. 20
- C. 80
- D. 100

13. 以下程序的输出结果是 ()

```
#include<iostream>
using namespace std;
void func( int a, int b, int& rc)
{
    rc = a * b;
}
int main()
{
    int c=4;
    func(3, 3, c);
    func(c, c, c);
    cout<<c;
}
```

- A. 4
- B. 9
- C. 16
- D. 81

14. 以下程序的输出结果是 ()

```
#include<iostream>
using namespace std;
int func(int a)
{
    if (a==1)
        return 1;
    else
        return func(a-1)+1;
}
void main()
{
    cout<<func(3)<<endl;
}
```

- A. 1
- B. 3
- C. 5
- D. 8

15. 有一个函数 `int sum(int arr[], int n);` 有 `int b[4][3];`对 `sum` 调用正确的是 ()

- A. `sum(b,12)`
- B. `sum(&b[0],12)`
- C. `sum(&b[0][0],12)`
- D. `sum(b[0][0],12)`

16. 下列函数原型声明中，不正确的是 ()

- A. `double RectArea(double length, double weight);`
- B. `double RectArea(double length=1.0, double weight);`
- C. `double RectArea(double length, double weight=1.0);`
- D. `double RectArea(double length=1.0, double weight=1.0);`

17. 下列关于内联函数说法不正确的是 ()

- A. 内联函数在运行时将该函数的目标代码插入每个调用该函数的地方
- B. 类声明时定义的成员函数是内联函数
- C. 短小的函数适合采用内联函数的形式
- D. 内联函数体内不能包含 `switch` 语句

18. 以下能正确构成重载的是 ()

- A. `int max(int x, int y);`
 `int max(int y, int x);`
- B. `double fun(int x);`
 `int fun(int x);`
- C. `void show();`
 `void show(int n);`
- D. `double fun();`
 `double fun(int n=0);`

19. 设有 `int a=2, b=1, *p=&a;` 则执行以下语句 `*p += a+b;` 后, `a` 的值为 ()

- A. 2
- B. 3
- C. 4
- D. 5

20. 下列能正确进行字符串赋值的语句是 ()

- A. char str = "hello!";
- B. char str[6] = "hello!";
- C. char str[100]; str = "hello!";
- D. const char* str = "hello!";

21. 若有以下定义, 则下列语句执行后能显示 2 的是 ()

int table[2][4]={ {4,1,3,5},{25,42,2,91} }, *p=&table[0][0];

- A. cout<<table[2][3]<<endl;
- B. cout<<p[1][2]<<endl;
- C. cout<<*p+6<<endl;
- D. cout<<*(p+6)<<endl;

22. 若有以下定义 char str[]="abcdef",*sp1=str; 则执行 cout<<sp1+3;显示为 ()

- A. def
- B. 一个 32 位地址值
- C. d
- D. 100

23. 若有如下定义, 则下列函数调用在运行时不会出现错误的是 ()

char* str1="ABC", *str2="123",str3[]="ABC",str4[]={ '1','2','3' };

- A. strcpy(s1, s2);
- B. strcpy(s3, s4);
- C. strcpy(s3, s2);
- D. strcpy(s1, s4);

24. 下列能正确申请一个动态字符数组的是 ()

- A. char p=char new[10];
- B. char *p=new char[10];
- C. char p=new char;
- D. char *p=new char;

25. 下列有关类和对象的说法中, 不正确的是 ()

- A. 类是一种用户自定义的数据类型
- B. 对象是类的一个实例
- C. 进行类的声明时, 要为其数据成员分配存储空间
- D. 类的成员默认是 private 的

26. 下列对构造函数描述正确的是 ()

- A. 构造函数可以重载
- B. 定义对象时如果没有参数，则不执行构造函数
- C. 在同类对象相互赋值 (“=”号赋值) 时，不执行构造函数而执行复制构造函数
- D. 复制构造函数可以有多个形参

27. Test 为类类型，若有以下函数定义，当执行该函数中的 return 语句返回主调函数时，
下列说法正确的是 ()

```
void func(Test t)
```

```
{  
    .....  
    return;  
}
```

- A. 系统会调用 Test 类的构造函数
- B. 系统会调用 Test 类的复制构造函数
- C. 系统会调用 Test 类的析构函数
- D. 系统不会调用 Test 类的任何函数

28. 以下程序的输出结果 ()

```
class A  
{  
public:  
    A(int ia=1) { a=ia; cout<<a; }  
    ~A(){ cout<<a; }  
private:  
    int a;  
};  
void main() { A arr[2]={ A(0) };    cout<<"*"; }
```

- A. 0*
- B. 01*
- C. 01*01
- D. 01*10

29. 关于基类和派生类，下列说法正确的是 ()

- A. 派生类只能有一个直接基类
- B. 当派生类对象消亡时，一定会调用基类的析构函数
- C. 只有选择公有继承方式，才允许派生类的成员函数访问基类的保护成员
- D. 如果派生类的构造函数没有初始化列表，则不会调用基类的构造函数

30. 类的（ ）成员不允许通过类的对象直接访问，但允许其派生类的成员函数直接访问

- A. 公有
- B. 保护
- C. 私有
- D. 公有和保护

三. 判断题（共 15 分）

- 1、定义数组 `int A[10]={1,2,3,4,5};`，那么 `A[5]`的值是随机的。（ ）
- 2、定义数组 `int A[10]={1,2,3,4,5};`，那么 `A[10]`的值为 0。（ ）
- 3、定义数组 `int A[10]={1,2,3,4,5};`，那么 `A[10]`代表整个数组。（ ）
- 4、定义数组 `char s[10]={65,66,67,68,69};`，则执行输出语句 `cout<<s<<endl;`会显示：ABCDE。（ ）
- 5、定义数组 `char s[10]={“ABCDE”};`，该定义可以简化写作 `char s[10]={ABCDE};`（ ）
- 6、函数由函数名、函数类型、形式参数表、函数体 4 部分组成。（ ）
- 7、函数原型就是指函数类型。（ ）
- 8、程序中标识符的作用域按所处的位置可以分为：块作用域、函数作用域、类作用域、文件作用域。（ ）
- 9、函数的形式参数表不能为空。（ ）
- 10、函数体里必须有 `return` 语句，不能省略。（ ）
- 11、函数体里只能写一个 `return` 语句。（ ）
- 12、任何一个函数必须有返回值。（ ）
- 13、内层标识符与外层标识符同名时，内层标识符可见，外层标识符不可见。（ ）
- 14、`int *p = new int(10);`该语句从堆内存申请了 10 个整型数的存储空间。（ ）
- 15、派生类以 `public` 继承方式从基类继承的 `private` 成员，可以在派生类作用域范围内被访问。（ ）

四. 补充程序题（共 12 分）

1、猴子吃桃问题：猴子第一天摘下若干个桃子，当即吃了一半，还不过瘾，又多吃了一个。第二天早上又将剩下的桃子吃掉一半，又多吃了一个。以后每天早上都吃了前一天剩下的一半再加一个。到第 5 天早上想再吃时，见只剩下一个桃子了。求第一天共摘了多少。

（每空 2 分，共 6 分）

```
#include <iostream>
using namespace std;
void main()
{
    int day, x1, x2;
    day = 5;
    x2 = 1;
    while (day > 0)
    {
        ① _____//第一天的桃子数是第2天桃子数加1后的2倍
        ② _____
        ③ _____//修改循环控制变量
    }
    cout << "第一天摘了" << x1 << "个桃子。" << endl;
}
```

2、下列程序的功能是将字符串 s 中的所有空格删除，并输出 s，请填空完成程序。（每空 2 分，共 6 分）

```
#include<iostream>
using namespace std;
int main()
{
    char s[]="We are learning C++";
    int i=0, j=0;
    while(s[i]!='\0')
    {
        if(s[i]!=' ') ① _____;
        i++;
    }
    ② _____;
    cout<< ③ _____<<endl;;
    return 0;
}
```

五. 读程序题（共 20 分）

要求：读程序，把代码补充完整，并回答问题

```
#include <iostream>
```

```
#include<string>
```

```
using namespace std;
```

```
class animal
```

```
{
```

```
    private:
```

```
        int leg;
```

```
        string color;
```

```
    public:
```

```
        animal(int l,  string c)
```

```
        {
```

```
            leg = l;
```

```
            color = c;
```

```
            cout << "animal 出场了" << endl;
```

```
        };
```

```
        int getLeg() { return leg; };
```

```
        string getColor() { return color; };
```

```
        ~animal()
```

```
        {
```

```
            cout << "animal 消失了" << "\n";
```

```
        }
```

```
};
```

```
class whale: _____//①补充代码，使类 whale 以 public 的方式继承类 animal
```

```
{
```

```
    private:
```

```
        string name;
```

```
        string wishes = "爱护环境,人人有责! ";
```

```
    public:
```

```
        _____//②请补充完成 whale 类的构造函数
```

```
        {
```

```
            name = n;
```

```
            cout << getColor() << "的鲸鱼" << n << "出场了" << endl;
```

```
        };
```

```
        string getName() { return name; };
```

```

void whaleSays() { cout << wishes << endl; };
~whale()
{
    cout << "鲸鱼"<< getName() <<"消失了" << '\n';
}
};
int main()
{
    whale Blue_whale(0, "BLUE","Peace");
    cout<<"蓝鲸的颜色是: "<< Blue_whale.color<<endl; ③
    cout << "蓝鲸说: ";
    Blue_whale.whaleSays() ;
    return 0;
}

```

回答问题:

语句③有错误吗? _____

如有错误请改正: _____ ④

写出程序的运行结果:

- ⑤ _____
- ⑥ _____
- ⑦ _____
- ⑧ _____
- ⑨ _____
- ⑩ _____

北京邮电大学 2020——2021 学年第一学期

《C/C++程序设计 with 编程方法》期末考试试题（A）

考试 注意 事项	一、学生参加考试须带学生证或学院证明，未带者不准进入考场。学生必须按照监考教师指定座位就坐。 二、书本、参考资料、书包等物品一律放到考场指定位置。 三、学生不得另行携带、使用稿纸，要遵守《北京邮电大学考场规则》，有考场违纪或作弊行为者，按相应规定严肃处理。 四、学生必须将答题内容做在试题答卷上，做在试题及草稿纸上一律无效。 五、学生的姓名、班级、学号、班内序号等信息由教材中心统一印制。								
考试 课程				考试时间		年 月 日			
题号	一	二	三	四	五	六	七	八	总分
满分	23	30	15	12	20				
得分									
阅卷 教师									

一. 填空题：(每空 1 分，共 23 分)

- 1、目标（或 obj） 连接（或 link，或 build，或链接）
- 2、4 5
- 3、0xFF0000FF
- 4、34
- 5、abc e
- 6、15 30
- 7、0 102
- 8、void swap(int & , int &) void swap(int *, int *)
- 9、Beijing e S
- 10、构造 析构
- 11、private protected public private

二. 选择题（每题 1 分，共 30 分）

1-5 答案: DDCCC
6-10 答案: BCBDA
11-15 答案: AADBC
16-20 答案: BACDD
21-25 答案: DACBC
26-30 答案: ACDBB

三. 判断题 (共 15 分)

1-5 答案: 错错错对错
6-10 答案: 对错对错错
11-15 答案: 错错对错错

四. 补充程序题 (共 12 分)

1、猴子吃桃问题: 猴子第一天摘下若干个桃子, 当即吃了一半, 还不瘾, 又多吃了一个。第二天早上又将剩下的桃子吃掉一半, 又多吃了一个。以后每天早上都吃了前一天剩下的一半再加一个。到第 5 天早上想再吃时, 见只剩下一个桃子了。求第一天共摘了多少。

(每空 2 分, 共 6 分)

```
#include <iostream>
using namespace std;
void main()
{
    int day, x1, x2;
    day = 5;
    x2 = 1;
    while (day > 0)
    {
        ① x1 = (x2 + 1) * 2; /*第一天的桃子数是第2天桃子数加1后的2倍*/
        ② x2 = x1;
        ③ day--;
    }
    cout << "第一天摘了" << x1 << "个桃子。" << endl;
}
```

2、下列程序的功能是将字符串 s 中的所有空格删除, 并输出 s, 请填空完成程序。(每空 2 分, 共 6 分)

```
#include<iostream>
using namespace std;
```

```

int main()
{
    char s[]="We are learning C++";
    int i=0,j=0;
    while(s[i]!='\0')
    {
        if(s[i]!=' ') ①_____ ;
        i++;
    }
    ②_____ ;
    cout<< ③_____ <<endl;;
    return 0;
}

```

答案：

- ① s[j++]=s[i]
- ② s[j]='\0';或 s[j]=NULL
- ③ s 或&s[0]

五. 读程序题（共 20 分）

读程序，把代码补充完整，并回答问题

```

#include <iostream>
#include<string>
using namespace std;
class animal
{
    private:
        int leg;
        string color;
    public:

        animal(int l,  string c)
        {
            leg = l;
            color = c;
            cout << "animal 出场了" << endl;
        };
        int getLeg() { return leg; };
        string getColor() { return color; };

```

```

        ~animal()
        {
            cout << "animal 消失了" << "\n";
        }
};

class whale :public animal //①补充代码，使类 whale 以 public 的方式继承类 animal
{
    private:
        string name;
        string wishes = "爱护环境,人人有责! ";
    public:
        whale(int l, string c, string n) :animal(l,c ) //②请补充完成 whale 类的构造函数
        {
            name = n;
            cout << getColor() << "的鲸鱼" << n << "出场了" << endl;
        };
        string getName() { return name; };
        void whaleSays() { cout << wishes << endl; };
        ~whale()
        {
            cout << "鲸鱼"<< getName() <<"消失了" << "\n";
        }
};

int main()
{
    whale Blue_whale(0, "BLUE","Peace");
    cout<<"蓝鲸的颜色是： "<< Blue_whale.color<<endl; ③
    cout << "蓝鲸说： ";
    Blue_whale.whaleSays() ;
    return 0;
}

```

回答问题：

语句③有错误吗？ 有

如有错误请改正： cout<<"蓝鲸的颜色是： "<< Blue_whale.getColor()<<endl; ④

写出程序的运行结果：

⑤ animal 出场了


⑥ BLUE 的鲸鱼 Peace 出场了

⑦ 蓝鲸的颜色是： BLUE

⑧ 蓝鲸说：爱护环境,人人有责！

⑨ 鲸鱼 Peace 消失了

⑩ animal 消失了

 选择Microsoft Visual Studio 调试控制台

```
animal出场了  
BLUE的鲸鱼Peace出场了  
蓝鲸的颜色是：BLUE  
蓝鲸说：爱护动物,人人有责！  
鲸鱼Peace消失了  
animal消失了
```