

题号	一	二	三	四	五	六	七	八	九	十	总分	阅卷人
得分												

得分	阅卷人

一、单项选择题（每题 2 分，共 20 分，答案填在下面表格中）

1	2	3	4	5	6	7	8	9	10

1. 语句 `cout << 'C' << 'C'+32`; 输出的结果是:

- A. CC32 B. Cc C. 6799 D. C99

2. 关于 C++ 函数的下列描述，正确的是:

- A. 有些函数定义即使加上了 `inline` 关键词，编译程序也不会把它作为内联函数来对待;
B. 函数 `void f(int a){}` 与 `void f(int b=10){}` 是重载关系;
C. `void f(){ void g();}` 是在函数 `f()` 内部嵌套的定义了另一个函数 `g()`;
D. 假设有宏定义 `#define max(a,b) a>b?a:b`, 则 `cout<<max(2+2, 3)`; 的结果是 4。

3. 下面关于 C++ 数组的初始化，语法正确的是:

- A. `int a[3][2]={0,1,3},{5}};` B. `int a[][2]={1,2,3,4,5};`
C. `int a[3][] = {1,2,3,4};` D. `int a[2][3] = { 0, 1, 2, 3, 4, 5, 6 };`

4. 假设有 `int *p, a`; 则下面的语句错误的是:

- A. `p = a;` B. `p = new int;` C. `p = new int(6);` D. `p = new int[6];`

5. 不正确的表述是:

- A. 基类中的纯虚函数只有定义，没有实现
B. 如果一个类中包含没有实现的纯虚函数，则不能用这个类创建对象
C. 纯虚函数应当是非 `private` 成员函数
D. 一个类中，如果有 `virtual` 修饰的函数，则这个类是抽象类

6. 描述正确的是:

- A. `#include <xx.h>`, 编译器会首先到当前 `cpp` 所在的用户目录中寻找 `xx.h`
B. `#include "xx.h"`, 编译器不会搜索编译器或者环境变量指定的系统目录
C. 如果几个目录中都有 `xx.h`, `#include "xx.h"` 最多只会搜索到第一个就停止继续搜索
D. `#include <iostream>` 是引入了一个标准 C 形式而非 C++ 形式的头文件

7. 下面小段程序，哪个是正确的:

- A. `char * p = new int; * p = 'a'; delete p;`
B. `int *p = new int[25]; p[10] = 100; delete p;`
C. `char * p = new char[10]; p[0] = 'K'; delete [] p;`
D. `char * p = new int; char a[10]; p =a; delete []p;`

8. 如果限制 `a.cpp` 中的函数 `f()` 不能在其他 `cpp` 文件中被直接调用，则应该在 `f` 前面添加关键字

- A. `private` B. `static` C. `extern` D. `const`

9. 对于如下定义: `int k, a[10], *p=a`; 哪个表达式是不对的

- A. `*p++=2` B. `p++` C. `a++` D. `a[0]=k`

10. 以下说法中正确的是:

- A. 一个类一定会有无参构造函数
B. 构造函数的返回值类型是 `void`
C. 一个类只能定义一个构造函数，但可以定义多个析构函数
D. 一个类只能定义一个析构函数，但可以定义多个构造函数

得分	阅卷人

二、阅读程序, 写出运行结果（每 5 分，共 30 分，答案写在方框内）

```
1.
int main(){
    int a = 3, b = 4;
    int *p1=&a, *p2=&b;
    p1=&a; p2=&b;
    cout << a << b << endl;
    if (a<b){
        int* p=p1; p1=p2; p2=p;
    }
    cout << a << b << endl;
    cout<<*p1<<*p2<<endl;
}
```

```
2.
int a[] = {0 ,1, 2, 3};
int &b = a[1];
int *p = &b;
cout << a[3] << endl;
cout << b++ ;
cout << *p++ << endl;
cout << *p << *(p+1) << endl;
```

```
3.
#include "iostream"
using namespace std;
int main(void)
{   int  a[6], i;
    for (i=1; i<6; i++)
        {   a[i]=9*(i-2+4*(i>3))%5 ;
            cout<<a[i]<<'t';
        }
}

4.
class Obj {
public:
    Obj()
    {
        count++;
        cout <<"obj " << count << "remain" << endl;
    }
    ~Obj() {
        count--;
        cout <<"Obj " << count <<"left" << endl;
    }
    static int count;
};
int Obj::count = 0;
int main(){
    Obj c1;
    Obj *p = new Obj();
    delete p;
    Obj c2(c1);
    //提示，此处调用缺省复制构造函数
    return 0;
}
```

```
5.
class A{
    int a;
public:
    A() : A(0){ }
    A(int i):a(i) { cout << a; };
};
class B:public A{
    A a1,a2;
    int b;
public:
    B(int i):A(i+1),a1(i+2),a2(),b(i){
        cout << b <<endl;
    }
};
int main(){  A a(6);   B b(6);}
```

```
6、
#include <iostream>
using namespace std;
class Mammal
{
public:
    Mammal():itsAge(1) { cout << "Mammal constructor"<<endl; }
    virtual ~Mammal() { cout << "Mammal destructor"<<endl; }
    virtual void Speak() const { cout << "Mammal speak!"<<endl; }
private:
    int itsAge;
};
class Dog : public Mammal
{
public:
    Dog() { cout << "Dog constructor"<<endl; }
    ~Dog() { cout << "Dog destructor"<<endl; }
    void Speak() const { cout << "Woof!"<<endl; }
};
int main( )
{
    Mammal *pDog = new Dog;
    pDog->Speak();
    delete pDog;
    return 0;
}
```

得分	阅卷人

三、编写程序。（20 分）

1. 用递归方法求 n 阶勒让德多项式的值,递归公式为

$$P_n(x) = \begin{cases} 1, & n = 0 \\ x, & n = 1 \\ \frac{(2n - 1) * x * P_{n-1}(x) - (n - 1) * P_{n-2}(x)}{n}, & n > 1 \end{cases}$$

2. 编写程序输出 100 到 200 之间的所有素数。（所谓素数是指除了 1 和它本身外，不能被其他数所整除的数）

得分	阅卷人

四、编写程序（10 分）

要求程序输出结果: 7,6,5,5, 请编写类 CDemo

```
int main(){
    CDemo d(7);
    cout << (d--) << ",";
    cout << d << ",";
    cout << (--d) << ",";
    cout << d << endl;
    return 0;
}
```

得分	阅卷人

五、编写程序（10 分）

定义类模板 Point，有两个坐标(x, y)，私有成员 x 和 y 的类型可以不同，坐标的数据类型可以是 int、float, double 类型. 具有获取

坐标的成员函数 getX 和 getY，支持如下使用

```
Point<int, int> p1(10, 20);
Point<int, float> p2(10, 20.5f);
cout<<p1.getX()<<","<<p1.getY()<<endl;
cout<<p2.getX()<<","<<p2.getY()<<endl;
```

得分	阅卷人

六、补充程序（10 分）

公交刷卡播报场景仿真：乘坐公交的人可能是老师、学生，也可能是工人、农民或者某个程序员。请通过类的继承，完善类 Human、Teacher 和 Student。在 main 函数中模拟上车买票的场景，车上上来两个人，一个是老师，另一个是学生，调用普通函数 GetOn 两次，分别输出“老师投币买票”、“学生刷卡买票”。

```
class Human // 定义 Human 类
{ public:
    _____// 1、买票接口函数
    {   cout<<"人买票。"<<endl; }
};

class Teacher _____ //2、派生老师类
{ public:
    void BuyTicket()
    {   cout<<"老师投币买票。"<<endl;   }
};

class Student _____//3、派生学生类
{public:
    void BuyTicket()
    {   cout<<"学生刷卡买票。"<<endl;   }
};

void GetOn(Human *h)
{
    _____; //4、调用 BuyTicket
}

int main() // 主函数中模拟上车买票的场景，车上上来两个人，一个是老师，另一个是学生
{
    Teacher * pt = new Teacher();
    Student * ps = new Student();
    GetOn(pt); // 第一个人是老师，投币买票
    GetOn(ps); // 第二个人是学生，刷卡买票

    _____; // 5、销毁对象
    return 0;
}
```