

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. 已知语句 unsigned int a = 6;int b = -20, c = -3; cout<< ((a+b)>6)<< ((a+c)==3);, 其输出结果为
A 11 B 01 C 10 D 00

2. 已知 int x=1;, 则执行语句 x=(x=1+2,x*2);后, x 的值为
A 1 B 2 C 5 D 6

3. 对数组名作函数的参数, 下面描述正确的是
A 数组名作函数的参数, 调用时将实参数组复制给形参数组
B 数组名作函数的参数, 调用时将形参数组复制给实参数组
C 数组名作函数的参数, 实参数组和形参数组共用一段存储单元
D 数组名作函数的参数, 不能改变主调函数中的数据

4. 以下关于复合数据类型的描述正确的是
A 一个整型值可以赋值给枚举类型变量
B 两个数组可以进行整体赋值
C 结构类型数据可以进行整体复制
D 联合类型为其每个成员都分配存储空间

5. 已知语句 int f(char*, int&); char str[10]; int k;, 则对函数 f 的正确调用形式为
A f(str, &k); B f(str, k); C f(str[10], k); D f(str[10], &k);

6. 以下关于操作符重载的描述正确的是
A 可以改变原操作符的优先级和结合性
B 可以定义原来没有的运算符
C 操作符重载和函数重载是完全不同的概念
D 操作符 new 必须作为静态的成员函数来重载

7. 下面类的定义中有 () 处错误

```
Class Test{  
Public:  
    void Test (int val) ;  
    ~ Test();  
Private:  
    int a=2.4;
```

```
Test temp;  
Test *ptemp;  
  
}  
A 1    B 2    C 3    D 4
```

8. 以下关于构造函数的描述错误的是
A 析构函数的调用顺序与构造函数的调用顺序完全相反
B 一个类只能定义一个析构函数, 但可以定义多个构造函数
C 创建包含成员对象的类的对象时, 先执行成员对象类的构造函数, 再执行本身类的构造函数的函数体
D 自定义拷贝构造函数默认调用成员对象的自定义拷贝构造

9. 已知 Atest 为抽象类, 下列说明语句正确的是
A Atest f(int); B Atest *p; C int f(Atest); D Atest atemp;

10. 类模板的类型参数可以作为
A 可以作为数据成员的类型
B 可以作为成员函数的参数类型
C 可以作为成员函数的返回值类型
D 以上三者皆可

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）
注意：为节省卷面，阅读的代码前面统一省略 #include <iostream> using namespace std;

```
1.  
void myConv(int res[][4],int n){  
    int kernel[2][2]={ 1,0,0,1};  
    for (int i = 1; i< n; i++){  
        for (int j = 1; j < 4; j++) {  
            res[i][j] =kernel[0][0] * res[i - 1][j - 1]  
                + kernel[0][1] * res[i - 1][j]  
                + kernel[1][0] * res[i][j - 1]  
                + kernel[1][1] * res[i][j];  
        }  
    }  
}  
void printOneRow(int m[][4],int n){  
    for (int i : m[n])  
        cout << i;  
    cout << endl;  
}
```

```
int main(){
    int m[4][4];
    for (int i = 0;i < 4;i++)
        for (int j=0; j<4; j++)
            m[i][j] = j+1;
    printOneRow(m,1);
    myConv(m,4);
    printOneRow(m,2);
}
```

```
2.
void strFunc(const char * p1, const char * p2, char * p3){
    int i;
    for (;p1[i] && p2[i]; i++)
        p3[i] = (p1[i]>p2[i])?p1[i]:p2[i];
    p3[i] = 0;
}
int main() {
    char* strs[] = {"2023","Cpp","Java"};
    char** p = strs;
    char b[80];
    cout << *p++ << endl;
    strFunc(*p, p[1], b);
    cout << b << endl;
}
```

```
3.
int i = 1;
static int j = 2;
void m(int &x, int y){
    int i = 2;
    static int j = 0;
    cout << i << j ++ << x++ << y++ << endl;
    cout << ::i ++ << ::j++ << endl;
}
int main() {
    int i = 2, j = 3;
    m(i,j);
    m(i,j);
    cout << i << j << endl;
}
```

```
4.
#define div(x,y) x / y
int sum(int x, int y) {return x + y;}
int diff(int x, int y) {return x - y;}
void show(int a, int b, int (*p)(int ,int )){
    cout << p(a,b) ;
}
template<class T>
T func(T& a, T &b){
    return 'a' + a - b;
}
int main() {
    int i = 4, j = 2;
    cout << div ( i + j, i - j) ;
    cout << char(func<int>(i, j)) ;
    show (i,j, sum);
    show (i,j, diff);
}
```

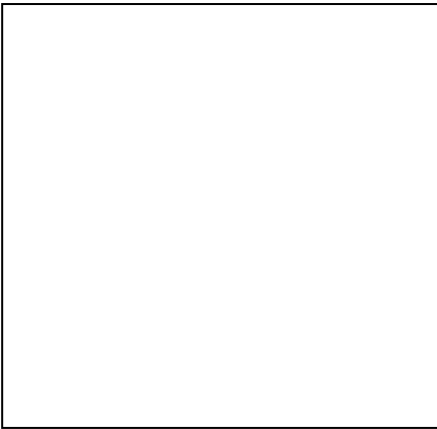
```
5.
class A {
private:
    static int count;
    int id;
public:
    A():id(++count) { }
    A(A& a):id(++count) { }
    A& operator=(A &b){ id = b.id; return *this;};
    void show(){ cout << count << id << endl;}
    virtual ~A() { count--; }
};
int A::count = 0;
A* m(A a1, A& a2){
    a1.show();
    a2.show();
    return new A();
}
```

```
int main(){
    A a1;
    A a2(a1);
    A *pa = m(a1, a2);
    pa->show();
    A c;
    c = a1;
    delete pa;
    c.show();
}
```

```
6.
class Point {
public:
    virtual void area()          { cout <<"Point Area " <<0 << endl; }
    void print(){ cout << "Point::print " << endl; }
    virtual ~Point()             { cout << "Point::~dtor " << endl; }
};
```

```
class Circle : public Point {
    int r = 2;
public:
    void area()          { cout <<"Circle Area " <<  r << endl; }
    void print(){ cout << "Circle::print " << endl; }
    ~Circle()            { cout << "Circle::~dtor " << endl; }
};
```

```
int main(){
    Point* pa = new Circle();
    pa->area();
    pa->print();
    Circle* pb =(Circle *) pa;
    pb->area();
    pb->print();
    delete pa;
}
```



得分	阅卷人

三、编写程序。（10 分）
编程实现下列功能：从键盘输入 x（弧度值），利用下列公式计算 cos(x)的近似值。要求准确度达到 10⁻⁵。

$$\cos x = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \frac{x^8}{8!} -$$

得分	阅卷人

四、编写程序。（10 分）

写一个完整的程序，从键盘读入两个浮点数和一个运算符，根据运算符计算两个浮点数的运算结果，并输出。运算符可能的值是 '+' , '-' , '*' , '/' 。

得分	阅卷人

五、编写程序。（10 分）

定义一个有理数类，实现下面的功能，为使程序健壮，至少使用 3 个 const。

```
int main()
{
    Rational a(4,5), b(7,3), c;
    cout << "a="; a.Display(); //输出 4/5
    cout << "b="; b.Display(); //输出 7/3
    c = a + b;
    cout << "a+b="; c.Display(); //输出 47/15
    c=-a;
    cout << "-a="; c.Display(); //输出 -4/5
    return 0;
}
```

得分	阅卷人

六、编写程序（10 分）

定义三个类 Person、Teacher、Student，实现下面功能：

```
int main()
{
    Teacher teacher("Shu Fen", 38, "Professor");
    Person * p;
    p = &teacher;
    p->show(); // 输出 Name: Shu Fen,age:38,title: Professor
    cout << endl;

    Student stu("Xi Jiajia", 20, 20230108);
    p = &stu;
    p->show(); // 输出 Name: Xi Jiajia,age: 20,stuid: 20230108
    cout << endl;
    return 0;
}
```

得分	阅卷人

七、编写程序（10 分）

定义一个类模板，并实现下列功能：

```
int main(){
    Point<int> A(4,5);
    cout << "A.x=" << A.Getx() << endl;

    Point<float> B(4.2,5.3);
    cout << "B.x=" << B.Getx() << endl;

    return 0;
}
```