

# 面向对象程序设计模拟试卷

## 一. 单选题(15)。

1. 关于定义 “struct A {int x; const int y=3;}; volatile A a={1};”, 如下叙述那个\_\_\_\_正确:  
A. a.x 是 int、a.y 是 const int 类型      B. a.x 是 volatile int、a.y 是 const int 类型  
C. a.x 是 int、a.y 是 const volatile int 类型      D. a.x 是 volatile int、a.y 是 const volatile int 类型
2. 关于 inline constexpr int &f(int &&a)的调用叙述\_\_\_\_正确:  
A. 该用常量做实参调用, 返回传统右值      B. 该用常量做实参调用, 返回传统左值  
C. 可用变量做实参调用, 返回传统右值      D. 可用变量做实参调用, 返回传统左值
3. 若派生类函数不是基类的友元, 关于该函数访问基类成员\_\_\_\_正确:  
A. 只有公有的可被派生类函数访问      B. 都可以被派生类函数访问  
C. 公有和保护的可被派生类函数访问      D. 都不对
4. 关于函数的所有缺省参数的叙述\_\_\_\_正确:  
A. 只能出现在参数表的最左边      B. 只能出现在参数表的最右边  
C. 必须用非缺省的参数隔开      D. 都不对
5. 对于定义 “char \* &&f();”, 如下哪个语句是错误的\_\_\_\_:  
A. f()=(char\*) "abcd";      B. \*f()= 'A';  
C. char \*p=f();      D. \*f()= "ABC"[1];

## 二. 在最多使用一级作用域 “::” 访问如 A::c 的情况下, 指出各类可访问的成员及其访问权限(20) 。

class A{ int a; protected: int b; public: int c; };	class C: A{ int g; protected: int h; public: int i; };
class B: protected A{ int d; protected: int e; public: A::c; int f; };	class D: B, C{ int j; protected: B::b; int k; private: int n; };

## 三. 指出 main 中每行的输出结果(20) 。

```
#include <iostream.h>
struct A{A(){ cout<<'A';}};
struct B{B(){ cout<<'B';}};
struct C: A{C(){ cout<<'C';}};
struct D: virtual B, C{D(){ cout<<'D';}};
struct E: A{
    C c;
    E(): c(){ cout<<'E';}}
```

```

};
struct F: virtual B, C, D, E{
    F(){ cout<<'F';}
};
void main(){
    A  a;  cout<<"\n";
    B  b;  cout<<"\n";
    C  c;  cout<<"\n";
    D  d;  cout<<"\n";
    E  e;  cout<<"\n";
    F  f;  cout<<"\n";
}

```

四. 指出以下程序的语法错误及其原因(15) 。

```

class A {
    static int  a = 0;
protected:
    int  b;
public:
    int  c;
    A(int) {};
    inline constexpr operator int() { return b; };
}  a(1, 2); //a=(1,2); //a=2
class B: A {
    B(int m){ b = m; };
    using A::b;
    virtual int d;
    int  e;
public:
    int b;
    friend B& operator =(const B& b) { return *this; }
    static B(int, int);
} b = 5;
class C: B {
public:
    C operator++(double) { return *this; };
}c;
int main( ) {
    int* A::* p, i;
    i = a.a;
    i = A(4);
    i = b.c;
    p = &A::c;
    i = b;
    return;
}

```

```
}
```

五. 指出 main 变量 i 在每条赋值语句执行后的值(15)。

```
int x=_____(请填入本人学号最后一位数字), y=x+30;
struct A{
    static int x;
    int y;
public:
    operator int(){ return x-y; }
    A operator ++(int){ return A(x++, y++); }
    A(int x::x+2, int y::y+3){ A::x=x; A::y=y; }
    int &h(int &x);
};
int &A::h(int &x)
{
    for(int y=1; y!=1|| x<201; x+=11, y++) if(x>200) { x-=21; y-=2;}
    return x-=10;
}
int A::x=23;
void main(){
    A a(54, 3), b(65), c;
    int i, &z=i, A::*p=&A::y;
    z=b.x;
    i=a.x;
    i=c.*p;
    i=a++;
    i::x+c.y;
    i=a+b;
    b.h(i)=7;
}
```

六. 为了没有误差地表示分数，定义分数类 FRACTION 用来表示分数，用整型 numerator 存分子、整型 denominator 存分母；并用\*重载分数约简运算、用>重载分数比较运算、用+重载分数加法运算、用\*重载分数乘法运算，以及相关的构造函数；可运用最大公约数函数 cmd 约简 (15)。

```
int cmd(int x, int y){
    int r;
    if(x<y){ r=x; x=y; y=r; }
    while(y!=0){ y=x%(r=y); x=r; }
    return x;
}
```