

C++模拟试卷-2016

一、选择题（每小题3分，共15分）

1. 类 A 是一个包含纯虚函数的抽象类，下列说明语法正确的是__C__

- A. A a;
- B. A f();
- C. A &f();
- D. A f(A);

2. 使用 exit 退出程序，关于对象自动析构，哪个叙述正确__B__

- A. 不析构全局对象但析构局部对象
- B. 析构全局对象但不析构局部对象
- C. 全局对象和局部对象都不析构
- D. 全局对象和局部对象都析构

3. 对于如下程序：

```
#include <stdio.h>
struct A {
    A() { printf("别理我"); }
    A(const char *s) { printf(s); }
} a("烦着呢");
A f() { printf("一边去"); return a; }
void main(void) { A f(); }
```

关于程序的输出，哪个叙述是正确的__A__

- A. 输出为“烦着呢”；
- B. 输出为“烦着呢 别理我”；
- C. 输出为“一边去”；
- D. 输出为“烦着呢 一边去”；

4. 对于 int x； 如下运算错误的是__A__

- A. x++ ++
- B. ++ ++x
- C. (++x)++
- D. ++ (++x)

5. 对于 int x， int &y 最好引用如下哪个表达式__A__

- A. x+=3;
- B. x+4;
- C. x++;
- D. (--x)--;

二、指出各类可访问的成员及其访问权限（20）。

```
class A {
    int a;
protected:
    int b, f;
public:
    int c, d;
};
```

A:
private: a
protected: b, f
public: c, d

```
class B: protected A {
    int d;
protected:
    int c, e;
public:
    int f;
};
```

B:
private: d
protected: c, e, A::(b, c, d, f)
public: f

```
class C: public A {
    int g;
protected:
    int h, d;
public:
    int c, i;
};
```

C:
private: g
protected: h, d, A::(b, f)
public: c, i, A::(c, d)

```
struct D: B, public C {
    int j;
protected:
    int k, c;
public:
    int n;
};
```

D:
private:
protected: k, c, B::(c, e, A::(b, c, d, f)), C::(h, d, A::(b, f))
public: j, n, B::(f), C::(c, i, A::(c, d))

三、指出 main 中每行的输出结果（前四题每题 3 分，后2题每题4分）

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

//struct E: virtual A, virtual D {

```
struct E: A, virtual D {
    D d;
    E(): A() { cout << 'E'; }
};
struct F: B, virtual C, E, virtual D {
    D d;
    F() { cout << 'F'; }
};
```

```
void main() {
```

```

A a; cout << "\n";    //输出= A
B b; cout << "\n";    //输出= B
C c; cout << "\n";    //输出= AC
D d; cout << "\n";    //输出= ACBD
E e; cout << "\n";    //输出= ACBDAACBDE
F f; cout << "\n";    //输出= ACBDBAACBDEACBDF
}

```

四、指出以下程序的语法错误及其原因（每错约 1分，共 15 分）

```

class A {
    int a;
protected:
    const int &b;
    ~A() {}
public:
    int c;
    virtual A (*g)(int);    //(1) g是一个变量，不能定义为virtual
    A(int x) { a = x; };    //(2) b没有初始化
} x = (4, 3);                //(3) x不能析构

class B: A {
    int d;
public:
    A::b;    //
    friend int operator()(int) { return 2; };    //(4) 不能声明为友元
    B(int x, int y, int z) { d = x + y + z; };    //(5) 没有说明基类的构造方法
} b(5, 6, 7);    //(6) 不能生成对象b

struct C: B {
    int z;
public:
    ~C(int x) { z = x; };
} c;    //(7) 没有构造函数；(8) 不能生成对象c

void main() {
    int A::*p = &c.z;    //(9) 成员指针不能指向物理地址
    int i = x.b;    //(10) 不能访问类A的保护成员
    i = x;    //(11) 类A没有强制类型转换函数
    i = b.b;    //
    i = i + c.d;    //(12) 不能访问类B的私有成员d
    i = b.*p;    //(13) p不是类A的成员指针
}

```

五、请填入学号最后一位十进制数字，指出 main 函数中变量 I 在每条赋值语句执行后的值（每小题 2.5 分，共 15 分）

```
int x = _____ //学号最后一位十进制数
int y = x + 3;
struct A {
    int x;
    static int &y;
public:
    operator int( ) const { return x + y; }
    int &v(int &x) {
        for(int y = 1; x < 301; x ^= y++) {
            if(x > 300) { x -= 31; y = 2; }
        }
        return ++x;
    }
    A &operator++() { ++x; ++y; return *this; }
    A(int x = A::y + 2, int y = ::x + A::y) { A::x = x + 1; A::y = y + 2; }
};
int &A::y = ::y;
void main() {
    A a(3, 4), b(a), c;
    int i, &j = i;
    int A::*p = &A::x;
    j = a.x;           // i =
    i = a.y;           // i =
    i = a.*p;          // i =
    i = ++a;           // i =
    i = b.y + ::y;     // i =
    (b.v(i) = 3) += 2;  // i =
}
```

答案（假设学号最后一位 = n）：

```
void main() {
    A a(3,4);           //a.x=4, ::y=6
    A b(a);             //b(a) 浅拷贝构造: b.x=4, ::y=6
    A c;               //c(::y+2,::x+::y)=c(8, n+6), c.x=9, ::y=n+8
    int i, &j = i;
    int A::*p = &A::x;
    j = a.x;           //i=4
    i = a.y;           //i=n+8
```

```

i = a.*p;           //i=4
i = ++a;           //++a: a.x=5,::y=n+9, i=n+14
i = b.y + ::y;      //i=2n+18
(b.v(i) = 3) += 2;  //i=5
};

```

分析:

	::x	::y, A::y	a.x	b.x	c.x	i, j
	n	n+3				
a(3, 4)		6	4			
b(a)		6		4		
c <=> c(8, n+6)		n+8			9	
j = a.x						4
i = a.y						n+8
i = a.*p						4
i = ++a		n+9	5			n+14
i = b.y + ::y						2n+18
(b.v(i) = 3) += 2						5

六、一个自然数，如果它等于除其本身之外的所有其它不同因子之和，则这个自然数被称为完美数。例如， $6=1+2+3$, $28=1+2+4+7+14$ 。试编写如下完美数类中的所有函数成员的函数体代码（每小题 2.5 分，共 15 分）

```

class PER {
    const int n;           //存放自然数
    int *const f;          //存放所有有效因子, 所有因子之和等于 n 才是完美数
    int c;                 //有效因子个数：正数表示是完美数，负数表示不是
public:
    PER(int p);            //用自然数 p 初始化 n, f, c
    PER(const PER &p);     //深拷贝构造函数
    PER &operator=(const PER &p); //深拷贝赋值运算
    operator int () const; //若不是完全数则返回 0，否则返回 c
    int operator [ ] (int k) const; //返回 k 所指示的因子, 若 k < 0 或 k >= c 返回 0
    ~PER();               //析构函数
};

```

//提示: f 分配的整型内存单元数量不会超过 $n/2$ 。

PER::PER(int p): n(p), f(new int [p])

```

{
    c = 0;
    for(int k = 1; k < n; k++)
    {
        if((n % k) == 0) f[c++] = k;
    }
    int sum = 0;
    for(int k = 0; k < c; k++) sum += f[k];
    if(sum != n) c = -1;
}

```

```

PER::~~PER()
{ if(f) { delete[] f; *(int *)&f = 0; } }

```

```

//深拷贝构造函数
PER::PER(const PER &p): n(0),f(0)
{ *this = p; }

```

```

//深拷贝赋值运算
PER &PER::operator=(const PER &p)
{
    this->~PER();
    *(int *)&n = p.n;
    *(int **)&f = new int[n];
    c = p.c;
    for(int k = 0; k < c; k++) f[k] = p.f[k];
    return *this;
}

```

```

//若不是完全数则返回 0， 否则返回 c
PER::operator int() const
{ return c > 0 ? c : 0; }

```

```

//返回 k 所指示的因子, 若 k < 0 或 k >= c 返回 0
int PER::operator[](int k) const
{ return k < 0 || k >= c ? 0 : f[k]; }

```