

这是期末考试的一部分，在这部分答题时，不能开启考试监考客户端以外的任何软件，包括但不限于浏览器、QQ、微信、支付宝、编程软件等

4-1 填空题 15 程序填空题 2

4-1 The output of the code below is:

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

class Str{
    char m_s[10];
    char *m_p;
public:
    Str(char *s){strcpy(m_s,s);m_p = m_s;}
    operator char*(){ return m_p; }
    char *operator++(){ return ++m_p; }
    char operator [](int i){ return m_s[i]; }
};

int main(){
    Str s("HEL");
    cout << s << endl;
    ++s;
    cout << s[0] << endl;
    cout << *s << endl;
}
```

The 1st line is (1分) H
The 2nd line is (1分) H
The 3rd line is (1分) i

4-2 程序填空题 2

4-2 The output of the code below is:

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

class A{
public:
    A(){cout << "A()" << endl;}
    ~A(){cout << "~A()" << endl;}
};

class B : public A{
public:
    B(){cout << "B()" << endl;}
    ~B(){cout << "~B()" << endl;}
};

int main(){
    A a;
    B b;
}
```

The 1st line is (1分) A()
The 2nd line is (1分) A()
The 3rd line is (1分) B()
The 4th line is (1分) ~A() X ~B() X
The 5th line is (1分) ~B() X ~A()
The 6th line is (1分) ~A()

4-3 程序填空题 2

4-3 The output of the code below is:

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

template <typename T>
class FF{
    T a1,a2,a3;
public:
    FF(T b1, T b2, T b3):a1(b1),a2(b2),a3(b3)
    {}
    T Sum() const
    {
        return a1 + a2 + a3;
    }
};

int main()
{
    FF<int> x(2,3,4),y(-2,-3,-4);
    cout << x.Sum() << endl << y.Sum() << endl;
}
```

The 1st line is (1分) 9
The 2nd line is (1分) -9

4-3 答案正确 (2分)

4-4 The output of the code below is:

```
#include <iostream>

struct A {
    A() { std::cout << "A" << std::endl; }
    A(const A &a) { std::cout << "B" << std::endl; }
    A& operator=(const A &a) { std::cout << "C" << std::endl; return *this; }
};

int main() {
    A a[2];
    A b = a[0];
    A c;
    c = a[1];
}
```

The 1st line is (1分) A
The 2nd line is (1分) A
The 3rd line is (1分) A B ?
The 4th line is (1分) A
The 5th line is (1分) C

4-4 答案正确 (2分)

4-5 The output of the code below is:

```
#include <iostream>

class C {
public:
    explicit C(int) {
        std::cout << "1" << std::endl;
    }
    C(double) {
        std::cout << "d" << std::endl;
    }
};

int main() {
    C c1(7);
    C c2 = 7;
}
```

The 1st line: (1分) 1
The 2nd line: (1分) d

4-5 答案正确 (2分)

4-6 The output of the code below is:

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

class A{
public:
    void F(int){ cout << "A::F(int)" << endl; }
    void F(double){ cout << "A::F(double)" << endl; }
    void F2(int){ cout << "A::F2(int)" << endl; }
};

class B : public A{
public:
    void F(double){ cout << "B::F(double)" << endl; }
};

int main(){
    B b;
    b.F(2.0);
    b.F(2);
    b.F2(2);
}
```

The 1st line is (1分) B::F(double)
The 2nd line is (1分) A::F(int) X
The 3rd line is (1分) A::F2(int)
B::F(double) X
A::F(int) X
A::F2(int)

4-6 答案正确 (2分)

4-7 The output of the code below is:

```
#include <iostream>

struct A {
    virtual void foo(int a = 1) {
        std::cout << "A" << '\n' << a;
    }
};

struct B : A {
    virtual void foo(int a = 2) {
        std::cout << "B" << '\n' << a;
    }
};

int main () {
    A *a = new B;
    a->foo();
}
```

The 1st line is (1分) B
The 2nd line is (1分) 2 1
B::foo
B::foo

4-7 答案正确 (2分)

4-8 The output of the code below is:

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

template<typename T>
T func(T x, double y){
    return x*y;
}

int main(){
    cout << func(2,7,3) << endl;
    cout << func(3,2,7) << endl;
}
```

The 1st line is (2分) 5.7
The 2nd line is (1分) 5
5.7 int, 7/5

4-8 答案正确 (3分)

4-9 The output of the code below is:

```
#include <iostream>

template<class T> void f(T &t) { std::cout << 1; }
template< void f(const int &i) { std::cout << 2; }

int main() {
    int i = 24;
    f(i);
}
```

The output is (1分) 1

4-9 答案正确 (1分)

4-10 The output of the code below is:

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

class A{
    int a[10];
public:
    int operator[](int i)const{
        cout << "operator[](int)const" << endl;return s[i];
    }
    int &operator[](int i){
        cout << "operator[](int)" << endl;return s[i];
    }
};

int main(){
    A a;
    const A &a2 = a;
    a[9] = a2[1];
}
```

The 1st line is (1分) operator[](int)const
The 2nd line is (1分) operator[](int)

4-10 答案正确 (2分)

4-11 The output of the code below is:

```
#include <iostream>
using namespace std;
class A{
public:
    static void f(double){
        cout << "f(double)" << endl;
    }
    void f(int){
        cout << "f(int)" << endl;
    }
};

int main(){
    const A a;
    a.f(3);
}
```

The output is (1分) f(double)

4-11 答案正确 (2分)

4-12 The output of the code below is:

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

class A{
public:
    virtual ~A(){}
};

class B : public A{};

int main()
{
    A a;
    B b;

    A *ap = &a;
    if (dynamic_cast<B*>(ap))
        cout << "OK1" << endl;
    else
        cout << "FAIL" << endl;
    if (static_cast<B*>(ap))
        cout << "OK2" << endl;
    else
        cout << "FAIL" << endl;

    ap = &b;
    if (dynamic_cast<B*>(ap))
        cout << "OK3" << endl;
    else
        cout << "FAIL" << endl;
    if (static_cast<B*>(ap))
        cout << "OK4" << endl;
    else
        cout << "FAIL" << endl;
}
```

The 1st line is (1分) FAIL
The 2nd line is (1分) FAIL OK2
The 3rd line is (1分) OK3
The 4th line is (1分) OK4

4-12 答案正确 (2分)

4-13 The output of the code below is:

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

class A{
public:
    static int m;
    A(int m,int n){
        this->m = m;
        this->n = n;
    }
    void print(){
        cout << m << "..." << n << endl;
    }
};

int A::m;

int main(){
    A a1(3,4);
    A a2(5,6);
    a1.print();
    a2.print();
}
```

The 1st line is (2分) 5...4
The 2nd line is (1分) 5...6

4-13 答案正确 (2分)

4-14 The output of the code below is:

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

class A{
public:
    A(){ cout << "A()" << endl; }
    A(const A&A){ cout << "A(const A&)" << endl; }
    A&operator=(const A&A){
        cout << "operator=(const A&)" << endl;
        return *this;
    }
};

int main()
{
    A a1,a2;
    a2 = a1;
    A a3 = a2;
}
```

The 1st line is (1分) A()
The 2nd line is (1分) A()
The 3rd line is (1分) operator=(const A&)
The 4th line is (1分) A(const A&)

4-14 答案正确 (2分)

4-15 The output of the code below is:

```
#include <iostream>

struct Base
{
    virtual ~Base()
    {
        std::cout << "Destructing Base" << std::endl;
    }
    virtual void f()
    {
        std::cout << "I'm in Base" << std::endl;
    }
};

struct Derived : public Base
{
    ~Derived()
    {
        std::cout << "Destructing Derived" << std::endl;
    }
    void f()
    {
        std::cout << "I'm in Derived" << std::endl;
    }
};

int main()
{
    Base *p = new Derived();
    (*p)-f();
    p->f();
    delete p;
}
```

The 1st line: (1分) I'm in Derived
The 2nd line: (1分) I'm in Derived
The 3rd line: (1分) Destruct... Derived
The 4th line: (1分) Destruct... Base

4-15 答案正确 (2分)