

**Scuola Superiore Sant'Anna**  
**International Master in Software Engineering**  
**Academic Year 2001-2002**

***Advanced Course on C++***

**TEST**

(code 1234)

**Name:** \_\_\_\_\_

**1) *What happens in the following code?***

```
MyClass a;  
MyClass *p;  
...  
p=a;
```

- a) **Compiler error**
- b) Linker error
- c) Run-time error
- d) None of the above

**2) *Consider the following code:***

```
class A {  
public:  
    void f(int a);  
    void f(int a, int b);  
    void f(double a);  
};  
...  
f(3.1, 4.5);
```

which function is called in the last line?

- a) It is a compiler error
- b) A::f(int)
- c) **A::f(int, int)**
- d) A::f(double)

**3) Consider the following code:**

```
int a = 1;
int &r = a;
int *p = &a;
...
cout << r++ << " - ";
cout << *(p++) << "\n";
```

what is printed on the screen?

- a) 1 - 1
- b) 2 - 2
- c) 1 - 2
- d) None of the above

**4) What is the output of the following code:**

```
void f(int a) {
    a++;
}
...
int i=0;
f(i);
cout << "i=" << i << "\n";
```

- a) i = 0
- b) i = 1
- c) Compiler error
- d) None of the above

**5) In the following list of sentences, some of them are correct and some are not. Check all the correct sentences:**

A copy constructor for an object **a** of class **A** is invoked when:

- a) passing a by value to function: f(A a);
- b) passing **a** by pointer to function: f(A \*p);
- c) passing **a** by reference to function: f(A &r);
- d) Initializing an object of type A: A b = a;
- e) Assigning **a** to another object b = a;

6) *In the following code, check the correct sequence of output on the screen:*

```
class A {
public:
    A() { cout << "(1)"; }
    A(A &a) { cout << "(2)"; }
};

class B {
    A a;
public:
    B() : a() { cout << "(3)"; }
};
...

B b;
B b2(b);
```

- a) (1)(2)(3)
- b) (1)(3)(2)**
- c) (3)(1)(2)
- d) None of the above

7) *Explain what is wrong with the following code:*

```
class A {
    class B {
        int i;
    public:
        B(int ii) : i(ii) {}
    };

    int j;
    B b;

public:
    A(int jj) : B(jj), j(jj+1) {}
    void f() { cout << b.i + j << "\n"; }
};
```

check the correct sentence:

- a) In function A::f() you cannot use an object of type B;
- b) In function A::f() you cannot access the variable b.i;**
- c) In the constructor of A you cannot use jj for initializing b;
- d) None of the above

**8) What is the output of the following code?**

```
class A {
public:
    A();
    void f() { g(); }
    virtual void g() { cout << "A::g()\n"; }
};

class B : public A {
public:
    void g() { cout << "B::g()\n"; }
};

class C : public B {
public:
    void f() { cout << "C::f()\n"; }
    void g() { cout << "C::g()\n"; }
};

...
A *p = new C;
p->f();
```

- a) C::f() and C::g()
- b) C::g()**
- c) B::g()
- d) A::g()

**9) Consider the following code:**

```
class A {
public:
    A() {}
    virtual void g() { cout << "A::g()\n"; }
};

class B : public A {
public:
    B() : A() {}
    virtual void g() { cout << "B::g()\n"; }
};

A a;
a.g();
```

what is the output?

- a) A::g()**
- b) B::g()
- c) Compiler error
- d) Run-time error

**10) Consider the following code:**

```
class A {...};  
class B : public A {...};  
...  
void f(A *pa);  
void f(B *pb);  
  
A *p = new B;  
f(p);
```

which function is invoked in the last line?

- a) **void f(A \*pa);**
- b) void f(B \*pb);

**11) Explain the advantages of passing by const reference instead of passing by value. Write a simple example for better explaining your ideas.**

**12) Consider the following code:**

```
class A {  
    int *pi;  
public:  
    A() { pi = new int; }  
};
```

What is wrong with this code? Explain the problem as clearly as possible and propose a possible solution.