外企笔试题精选二

1. 下面代码是否有错？如果有错，错在哪里？

```
struct Test
{
        Test() { }
        Test(int i) { }
        void func() { }
};
int main()
{
        Test t1(1);
        Test t2();
        t1.func();
        t2.func();
}
```

2. 下面的代码输出什么？为什么？

```
class Test
{
	int m_i;
	int m_j;
public:
	Test(int v) : m_j(v), m_i(m_j)
	{
	}
	int getI()
	{
		return m_i;
	}
	int getJ()
	{
		return m_j;
	}
};
int main()
{
	Test t1(1);
```

```
	Test t2(2);
	cout<<t1.getI()<<" "<<t1.getJ()<<endl;
	cout<<t2.getI()<<" "<<t2.getJ()<<endl;
}
```

3. 下面的代码输出什么？为什么？

```
class Test
{
	int m_i;
	int m_j;
public:
	Test()
	{
		cout<<"Test()"<<endl;
	}
	Test(int v)
	{
		cout<<"Test(int v)"<<endl;
	}
	~Test()
```

```
        {
                cout<<"~Test()"<<endl;
        }
};
Test Play(Test t)
{
        return t;
}
int main()
{
        Test t = Play(5);
}
```

4. Which virtual function re-declarations of the Derived class are correct?

A. Base* Base::copy(Base*);

Base* Derived::copy(Derived*);

B. Base* Base::copy(Base*);

Derived* Derived::copy(Base*);

C. int Base::count();

int Derived::count();

D. void Base::func(Base*) const;

void Derived::func(Base*);

5. 下面程序输出什么？为什么？

```
class Base
{
public:
	virtual void func()
	{
		cout<<"Base::func()"<<endl;
	}
};
class Child : public Base
{
public:
	void func()
	{
		cout<<"Child::func()"<<endl;
	}
};
```

```
int main()
{
	Base* pb = new Base();
	pb->func();
	Child* pc = (Child*)pb;
	pc->func();
	delete pc;
	pb = new Child();
	pb->func();
	pc = (Child*)pb;
	pc->func();
}
```

6. A C++ developer wants to handle a static _cast<char*>() operation for the String class shown below. Which of the following options are valid declarations that will accomplish this task?

```
class String
{
public:
// ...
// declaration goes here
```

};

A. char* operator char* ();

B. operation char*();

C. char* operator ();

D. char* operator String ();

7. 以下两种情况：

(1) new 一个 10 个元素的数组

(2) 分 10 次 new 一个整型变量

哪个占用的空间更大些？

A. 1

B. 2

C. 一样多

D. 无法确定

8. 下面程序输出什么？

```
int main()
{
        int v[2][10] =
        {
```

```
        { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10},
        {11, 12, 13, 14, 15, 16, 17, 18, 19, 20}
    };
    int (*a)[8] = (int(*)[8])v;
    cout<<**a<<endl;
    cout<<**(a + 1)<<endl;
    cout<<*(*a + 1)<<endl;
    cout<<*(a[0] + 1)<<endl;
    cout<<*a[1]<<endl;
}
```

9. 下面的程序输出什么？为什么？

```
class Base
{
public:
    int a;
    Base() { a = 1; }
    void println() { cout<<a<<endl;; }
};
class Child : public Base
```

```
{
public:
        int a;
        Child() { a = 2; }
};
int main()
{
        Child c;
        c.println();
        cout<<c.a<<endl;
}
```

10. 用 C/C++语言实现一个存储整形数据的栈数据结构。

要求实现以下功能：

(1) 入栈操作 push

(2) 出栈操作 pop

(3) 栈大小操作 size

(4) 栈中最小元素 min

11. 编程实现二叉树的相等比较，当二叉树每个结点中的值对应相等时，二叉树相等，否则不相等。

二叉树每个结点由如下结构体表示：

```
struct BTreeNode

{

int v;

BTreeNode* left;

BTreeNode* right;

};
```

函数原型：

```
bool BTreeCompare(BTreeNode* b1, BTreeNode* b2);
```