

## 条款11 理解特殊成员函数生成

① 特殊成员函数: a. 默认构造

b. 析构

c. 拷贝构造

d. 拷贝赋值

e. 移动构造

f. 移动赋值

编译器也可以自动生成

意义: 避免重复代码

② C++98 / C++11 法则

1° 析构函数 (2° 拷贝构造函数 3° 拷贝赋值函数) C++98  
4° 移动构造函数 5° 移动赋值函数 C++11

逐个拷贝/移动 non-static 数据

只要声明5中其中一个, 另外4个也应声明实现, 因为编译器不再合成

原因: 只有需要手动资源管理的类, 才会声明上述的函数, 并且

一般来说, 声明其中一个, 另外4个也不应该使用默认声明

③ 为兼容 C++98 ①/②/③ 声明其中几个, 另外几个编译器另会合成

④ C++11 的亡羊补牢

a. 声明①/②/③不会合成④/⑤

b. 声明④/⑤不会合成②/③/④/⑤ (析构一定会合成)

```
struct Base { virtual ~Base() { ... } }
```

```
Base (Base &&) = default; ... }
```

⑤移动构造/赋值更像一种请求,如果没有实现,

由于C++98(③),反而执行拷贝操作

```
class StringTable{
```

```
public: StringTable(){...}
```

```
~StringTable(){...}
```

```
private: std::map<int, string> value;};
```