

条款11 优先考虑使用delete函数而非使用未定义的私有声明

① C++98 如何禁用拷贝构造与拷贝赋值

将2个函数的声明写在 **private** 内且不去定义. 此时在类外调用时会出
现报错函数不可访问, 而不去定义是为了阻止友元类或成员函数调用, 此
时会出现链接报错(找不到定义), 但是不会报红.

② 使用 delete 完成操作

`A operator=(Const &A) = delete;`

通常 delete 掉的函数会写在 public 中, 不然报错信息不明确

③ delete 可用来删除任何函数

`bool isLucky(int number);`

`bool isLucky(char) = delete;`

`isLucky('a');` 报错

④ delete 也可禁止一些模板的实例化

`template<typename T>`

`void processPointer(T* ptr);`

`template<>`

可省略
介

`void processPointer<void>(void *ptr) = delete;`

尤其对于类中的函数模板

如果使用 C++98 写法会报错

class Widget {

public:

template<typename T>

void processPointer(T* ptr){...}

private:

template<>

void processPointer<void>(void*); }^{报错}

两者必须在同一访问级别