















**基础**

1. explicit构造函数

class Test

{

public:

explicit Test(int num);

};

explicit关键字阻止进行隐式类型转换，但仍可进行显式类型转换。

1. 拷贝构造函数和赋值构造函数

class Widget

{

public:

Widget();

Widget(const Widget& rhs);//构造函数

Widget& operator=(const Widget& rhs);//赋值构造函数

};

**条款01：视C++为一个语言联邦**

c++语言层次:

1. C.

C++仍是以c未基础。区块、语句、预处理器、内置数据类型、数组、指针

1. Object-Oriented C++

类、构造与析构、封装、继承、多态、动态绑定。。。

1. Template C++

模板元编程(TMP template metaprogramming)

1. STL

容器、迭代器、算法、函数对象

C++高效编程守则视状况变化，取决于使用C++哪一部分。

**条款02:尽量以const、enum、inline替换#define**

inline关键字:

inline关键字更多地出现在C中，很多短小的工作函数可以达到快速执行的效果。函数声明为inline表示编译器会将inline函数粘贴到调用点出，省去了原本函数查找、调用的时间。

使用规则:

1. 不可包含复杂的结构控制语句，包含循环控制和复杂的条件控制，如while、switch等

(2)不可包含递归函数

inline函数定义在头文件。

inline和宏函数的比较:

共同点:(1)都可以节省在函数调用方面带来的时间和空间开销。(2)实现的功能一致

区别:(1)在预编译时器，宏定义在调用处原样替换(宏展开)。在编译时期，内联函数在调用处展开，进行参数类型检查，宏定义不会进行参数类型检查。

(2)内联函数是函数，具有函数和宏定义的区别。

(3)内联函数可作为类的成员，可为类提供封装性，可以使用类的资源。

可以使用内联函数完全替代宏。

旧式编译器不允许static成员在其声明式上获得初值，可以采用将常量定义式为于实现文件内

enum hack:一个属于枚举类型(enumerated type)的是指可以权当ints被使用

class GamePlayer

{

private:

enum{NumTurns=5};

int scores[NumTruns];//数组初始化时必须明确数组长度

};

条款注意点:

* 对于单纯常量，最好以const对象或enum替换#define
* 对于形式函数的宏(macros)，最好改用inline函数替换#define

**条款03:尽可能使用const**

STL迭代器的作用如同T\*指针，声明迭代器为const就像声明指针为const一样，即:

const vector<T>::iterator== T\* const

const\_iterator表示迭代器指向的内容不能够被改变。

令函数返回一个常量值，往往可以降低因客户错误而造成的意外，而又不至于放弃安全性和高效性。