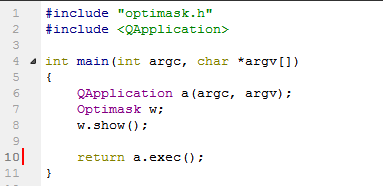
**1. main.cpp 在哪里被调用（没有看到），还是不需要被调用。**

**其输入参数 argc 和 \*argv 都是什么？**



（1）main函数是主程序入口函数，调用其他函数。

（2）对于[QT](https://www.baidu.com/s?wd=QT&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1YkujFBuHFbujfkPWD4ujfY0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EPHD4n1ndnHDvn1c4nHRsrj6d" \t "_blank)这样的图形编程来说，一般都是逻辑绘图分离的，QApplication代表一个程序，Window代表一个窗口（此时指Optimask）。argc和argv是命令行传进去的参数。Char \*argv[]是一个字符数组,其大小是int argc,主要用于命令行数 argv[] 参数，数组里每个元素代表一个参数;  
比如你输入    
   test   a.c   b.c   t.c    
   则    
   argc   =   4    
      
   argv[0]   =   "test"    
   argv[1]   =   "a.c"    
   argv[2]   =   "b.c"    
   argv[3]   =   "t.c"

argc记录了用户在运行程序的命令行中输入的参数的个数。    
arg[]指向的数组中至少有一个字符指针，即arg[0].他通常指向程序中的可执行文件的文件名。在有些版本的编译器中还包括程序文件所在的路径。  
因为图形编程有时也需要从命令行传递参数给程序，所以才会有argc和argv。

在调用一个可执行程序时，某些情况下需要向程序传递参数。如我们可以在控制台中键入notepad.exe，  
回车后将执行记事本程序。如果我们希望在打开notepad时同时打开一个文本文件，可以在notepad.exe    
后面跟上文件的路径和名字，如notepad.exe   example.txt（文件在当前路径）。     
  那么程序中如何能得到这些输入参数呢？这个工作是编译器帮我们完成的，编译器将输入参数的信息放入main函数的参数列表中。    
   main函数的参数列表保存了输入参数的信息，第一个参数argc记录了输入参数的个数，    
   第二个参数是字符串数组的，字符串数组的每个单元是char\*类型的，指向一个c风格字符串。    
   以notepad.exe   example.txt为例    
   argc是2，就是说argv数组中有两个有效单元    
   第一单元指向的字符串是"notepad.exe"    
   第二单元指向的字符串是"example.txt"      
   argv数组中的第一个单元指向的字符串总是可执行程序的名字，以后的单元指向的字符串依次是程序调用时的参数。      
  这个赋值过程是编译器完成的，我们只需要读出数据就可以了。

**2.optimask.cpp中， Optimask::Optimask(QWidget \*parent) : QMainWindow(parent)**

**那个 ： 是什么意思？**

**namespace是Optimask，其下的class也是Optimask，是这样理解吧？**





1. 两个冒号

::域名解析符  
返回值类型名 类名::函数成员名（参数表）   
{   
函数体。   
}   
这个是在类内声明函数后在类外定义的一种方法!  
如果不加"类名::"的话,编译系统就不会知道你的函数属于哪个类;  
另外,这样定义函数一定要在类中声明之后,说明它是[类的成员函数](https://www.baidu.com/s?wd=%E7%B1%BB%E7%9A%84%E6%88%90%E5%91%98%E5%87%BD%E6%95%B0&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1Ykn19WrH9Wm1ndujc3Phc40ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnWnsPWb1PHD" \t "_blank)才可以,  
在类内声明的时候就不需要::了,直接 返回值类型 函数名(参数表) 就可以了!

（2）一个冒号

类名冒号后面的是用来定义类的继承。  
class 派生类名 : 继承方式 基类名  
{  
派生类的成员  
};  
继承方式：public、private和protected，默认处理是public。

Optimask::Optimask(QWidget \*parent) : QMainWindow(parent)

表示此类Optimask继承QMainWindow类

（3） Optimask是类名，这个类继承了QMainWindow，它有自己的成员变量和成员函数都是自己声明和定义，它声明的对象可以直接使用父类QMainWindow的函数

**3. /Layer/gds2reader.cpp 其中 gds2reader.cpp可能在 /File 目录下好些。**

**4. 可否把 子槽函数放到对应的子目录下，这样optimask.cpp 就不会一直变大，而是可以基本稳定。比如void Optimask::file\*\*\*( ) { } 等函数放在 \File\ 目录， void Optimask::draw\*\*\*( ) { } 等函数放在 \Draw\ 目录， 如此类推？相应的UI file也对应在对应子目录下。**

可以

**7. Optimask::~Optimask( ){ } 这个 ～ 什么意思？这个函数目的是什么？**

**函数名是在类名前加上~，表明是析构函数**

类的析构函数，它是类的一个成员函数，名字由波浪号加类名构成。看它的名字我们大概就能联想到他功能，是执行与构造函数相反的操作：释放对象使用的资源，并销毁非static成员。

a.完成整个界面区域的编排。目前已经完成1.菜单；3.构层；4. 模块树结构区； 5. 主视图区。

i. 继续完成6. 辅助视图区（Navigator）, 7. 编程区，8. 报错和提示区， 9. 状态信息区。

ii. 每个区必须可以打开和关闭，拉大和缩小，移动位置，重新组合。

b. 图形与坐标数组对应。（就是画的任何一个图形，需要知道其对应的坐标组（X，Y），并且可以存储为坐标数组，比如(X0,Y0),(X1,Y1), ....., (Xn-1,Yn-1)的矩阵形式。这样用鼠标点击选择图形后，才有可能做任何布尔和变换操作。只有某个图形的坐标可存后，才可以结合数据结构进行后续工作）

c. 请学生学习GitHub的使用，以及代码同步（邓恒华有简介文档）。我们可以添加用户（前天微信图）。请尽快使用网络同步开发。

d. 请学生学习Optimask\_File\_Format.xmind，更好理解程序结构和需求。