**多边形布尔运算平台**

**开**

**发**

**技**

**术**

**文**

**档**

**2014213501 贾晨**

**2014年12月18日**

## 程序框架结构

### 模块划分

程序整体分为如下三个主要的模块：

* CP\_Polygon(.cpp/.h)：定义多边形数据结构并初始化，实现对多边形的一些基本操作，例如移动元素、添加元素、显示方式等；
* CP\_PolygonPlatformDoc(.cpp/.h)：主要用来完成保存、打开txt文档等功能；
* CP\_PolygonPlatformView(.cpp/.h)：实现程序界面的显示、程序功能的内部代码实现等。

### 程序设计整体思路

1. 数据结构

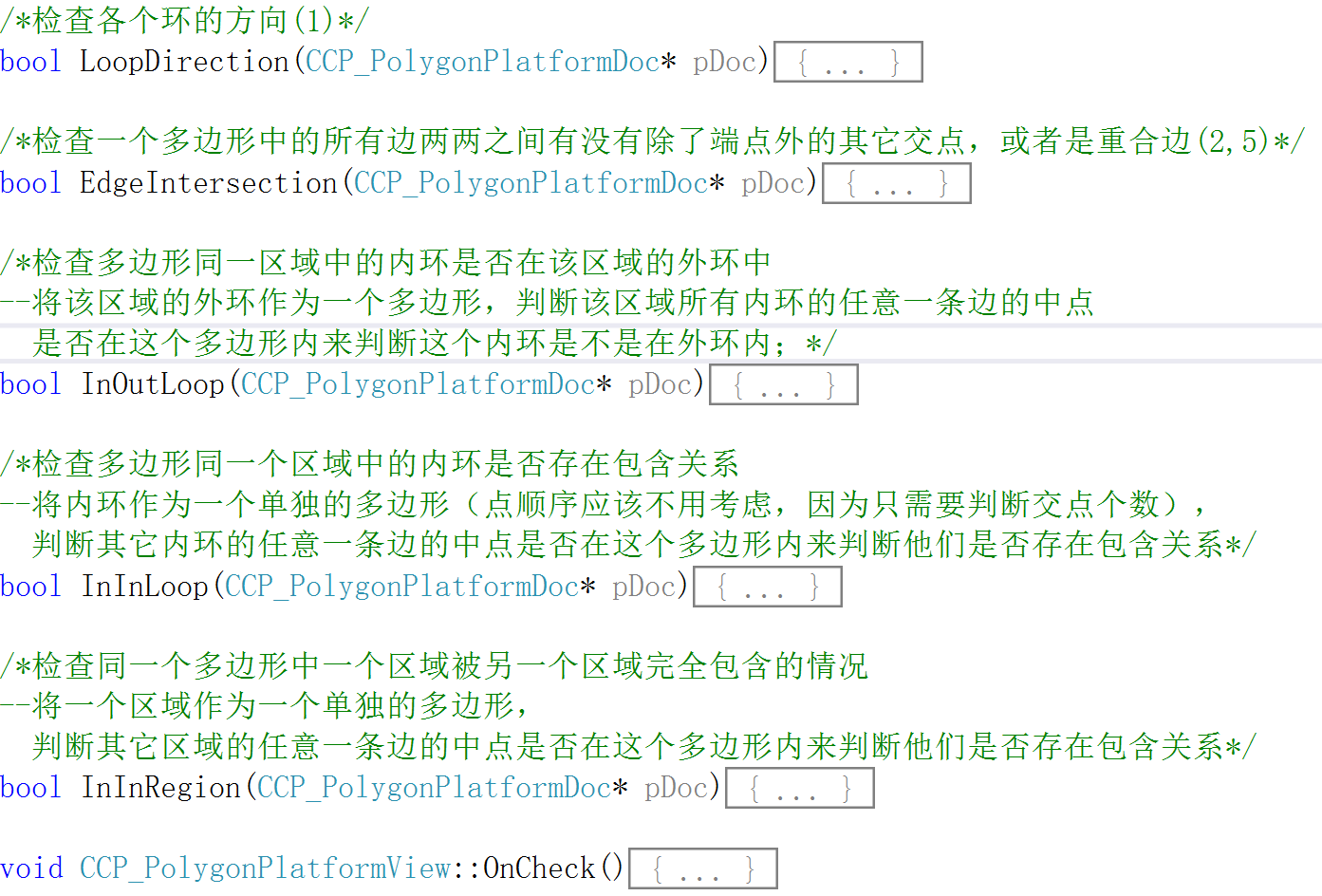
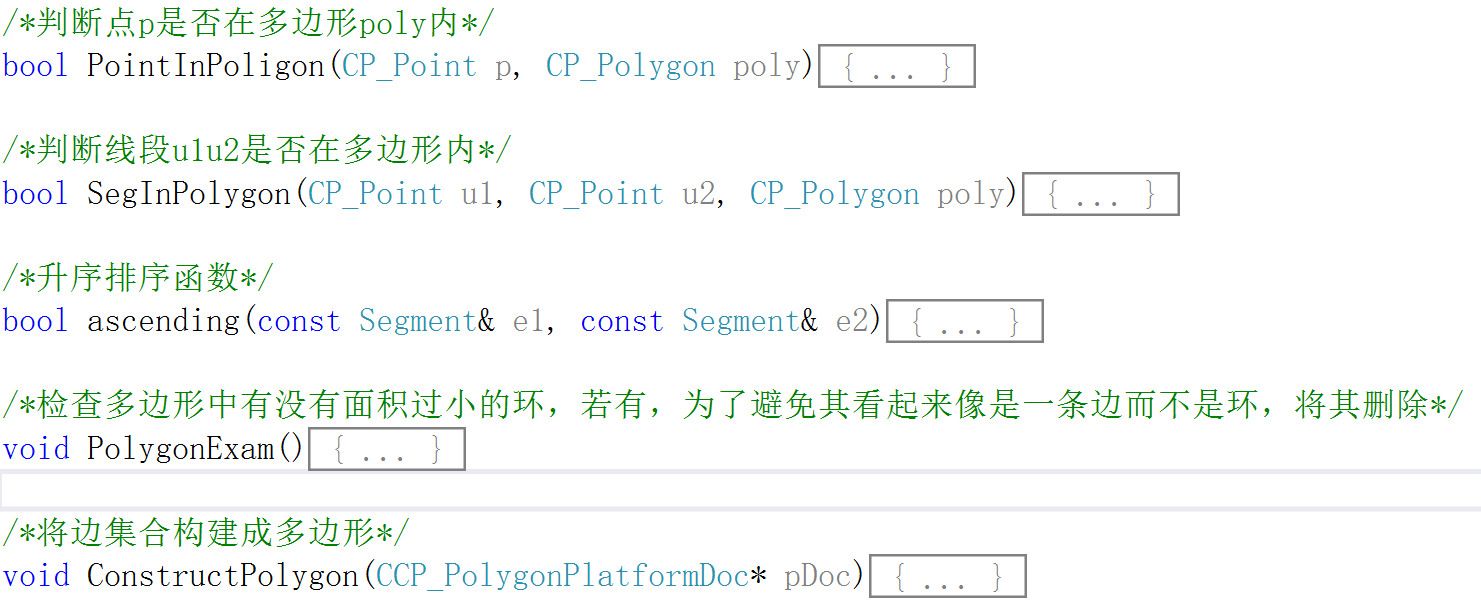
程序中定义了如下几种数据结构

1. 函数设计
2. 平台绘制函数

最初搭建的平台中就带有完整的绘制函数，故此处便不再赘述。

1. 功能实现函数

实现布尔运算平台涉及到的所有函数如下所示：

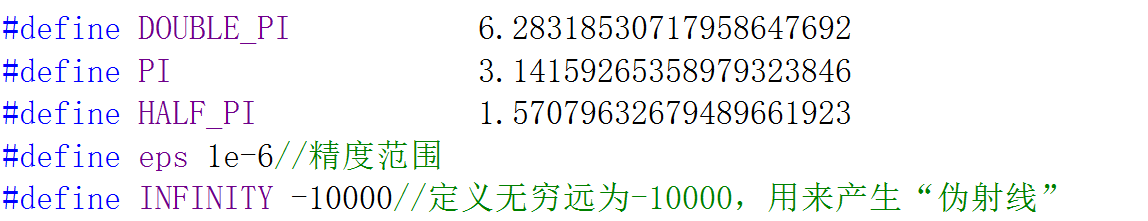


## 用到的一些技巧

### 编程技巧

1. 使用宏定义

对于常使用的常数，使用宏定义更加简洁、方便，且能避免出错：



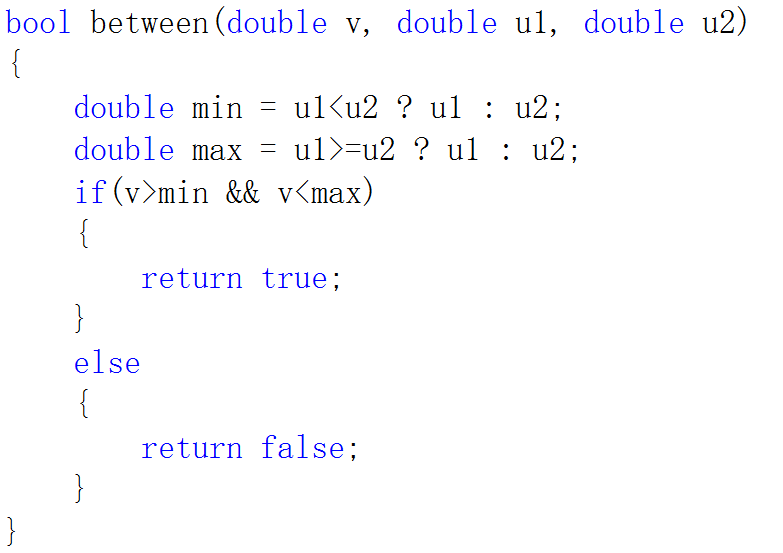
1. 处理容差

由于坐标是double型数据，存储和计算的过程中会出现偏差，所以“判零”或者“判等”时需要考虑容差，使用如下的方式可以方便简洁地处理容差问题：



1. 功能隔离

将一些常用的简单的，但是编码过程有些复杂的功能封装到一个独立的函数中，使用时直接调用函数，更加直观、简洁、方便。如下图所示函数，实现的功能为：判断u1<=v<=u2或u2<=v<=u1是否成立：



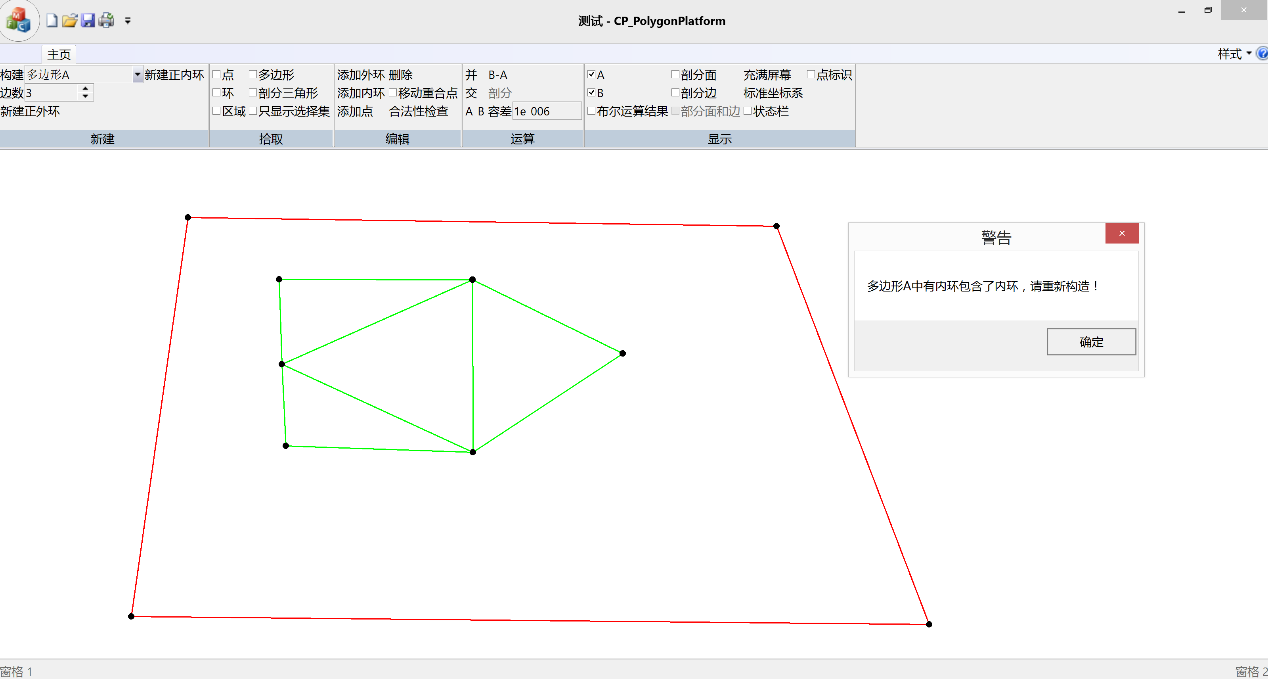
### 测试技巧

1. 代码重用测试

在代码开发的过程中有些模块功能几乎完全相同，此时应该先写好一个模块，测试完成保证基本无误后再移至到另一个模块上。同时要注意，重用代码时可能会出现忘记在原有代码上进行修改或修改不全面的情况，而造成错误，所以代码重用时要注重测试这些方面bug。

1. 边界测试

由于布尔运算过程中会出现一些边界情况，例如边重合、顶点重合等情况，此时可以通过手动调整txt文档中的参数来生成一些带有边界情况的用例，如下图所示，为两内环交点为顶点，但是内环相互重合的非法情况：



1. 认真做好测试记录

在做测试时及时记录测试数据，及测试结果。在修改代码后可以重新做一次测试，避免由于修改代码产生其它的不可预知的bug。

## 程序编译、配置及运行

此程序使用MFC和C++结合编写，运行环境为Windows操作系统下的VS2012，无需进行其它配置，简单易行。点击工具栏中的“本地Windows调试器”或菜单栏中的“调试”->“开始执行（不调试）”即可进行程序的编译运行，得到运行结果。