

# 实验三 交互式绘制多边形

转载

巷中人 于 2019-06-03 17:05:00 发布 2438 收藏 11

文章标签: [移动开发](#) [开发工具](#) [python](#)

原文链接: <http://www.cnblogs.com/amt0p/p/10968688.html>

版权

## visual studio2019实现交互式绘制多边形

这个实验……在网上找不到。孔令德的实验代码下载下来有密码，暴力破解没希望，只能自己写了。这个和实验二是紧密相连的。

### 实验实现的目标

在屏幕上点击一个点，松开移动鼠标，线就会跟着鼠标进行延长，再次点击就会形成新的点，两点之间形成线段，然后新的线继续跟着鼠标跑。

当鼠标经过起点的时候，直线自动连到起点，形成闭合多边形，并且完成颜色的填充。

所以相对实验二，我们要实现直线的动态显示（橡皮筋绘制技术）和填充多边形。其中填充多边形用到了双缓冲机制，避免绘图的过渡（瞬间上色）。

### 橡皮筋技术和用户交互

直线在鼠标的移动下像橡皮筋那样拉伸，就是橡皮筋技术。实现这个技术同样需要双缓冲机制。

下面的代码是网上找来的。[来源](#)

```
int isPress=0;
CPoint start;
CPoint end;
```

```
//将上述代码放在***View.cpp里面的开头部分（只要能在***View.cpp里面起控制全局的作用就行，例如放在#endif后面）
```

//在\*\*\*View.cpp里面用添加类向导将单击鼠标左键，鼠标移动映射为相应的函数。并且添加以下代码

```
void CXiangpijinView::OnLButtonDown(UINT nFlags, CPoint point)
{
    // TODO: Add your message handler code here and/or call default
    isPress=1;
    start=point;
    end=start;
    CView::OnLButtonDown(nFlags, point);
}

void CXiangpijinView::OnMouseMove(UINT nFlags, CPoint point)
{
    // TODO: Add your message handler code here and/or call default
    CDC *pDC=this->GetWindowDC();
    if (isPress)
    {
        pDC->SetROP2(R2_NOT);
        pDC->MoveTo(start);
        pDC->LineTo(end);
        pDC->MoveTo(start);
        pDC->LineTo(point);
        end=point;
    }
    CView::OnMouseMove(nFlags, point);
}
```

以上代码实现了画线的橡皮筋功能，效果类似于实验二，但是是系统调用的方法。

## 实现绘制多边形

可以在屏幕上画线了，就要绘制多边形。

当鼠标左击一次，就创建一个点，鼠标移动显示橡皮筋拉伸的线。再次点击，生成第二个点，线段生成，并生成新的橡皮筋线。

我们需要保存画的点和线。

```
//在view.h中建立一个CPoint
CPoint point[100];
```

如果要保存边，就要创建新的类，AET和Bucket类

虽然孔令德的实验代码没办法得到，但是他的课本课后习题的代码在网上有，其中就有“多边形有效边表填充算法”的实验，里面实现了AET和Bucket

文件我就不发出来了，最后看我的代码就可以。

使用AET和Bucket保存边，并且进行填充。

最后实现画图的功能，vs2019工程文件，代码不方便一一展示，就放在后面供大家下载参考吧。

程序刚启动的时候不要着急点，等待程序加载，不然你会发现点不出直线。  
左键用来画线，右键用来绘制，切记！！

代码：<https://share.weiyun.com/5ilVZUq>

转载于：<https://www.cnblogs.com/amttop/p/10968688.html>