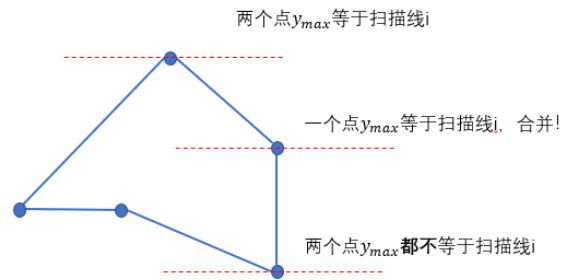


计算机学院实验报告

实验题目： 多边形的扫描转换与填充		学号： 202000130143			
日期： 10. 4	班级： 计科 20. 1	姓名： 郑凯饶			
Email: 1076802156@qq.com					
实验目的： 了解多边形的表示方式，区域填充基本原理，掌握多边形的扫描转换算法。					
实验环境介绍： Dell Latitude 5411 Intel(R) Core(TM) i5-10400H CPU @ 2.60GHz (8GPUs), ~2.6GHz Windows 10 家庭中文版 64 位 (10.0, 版本 18363) Visual Studio 2022					
解决问题的主要思路： 输入： 顺/逆时针存储多边形顶点集合 输出： 内部填充的多边形 通过扫描线算法进行填充。首先将顶点集合转化为边集合，存储于新边表 NET，对于相邻顶点 (x_i, y_i) ， (x_{i+1}, y_{i+1}) ，构造对应的边e表示： <table border="1" data-bbox="432 1189 1015 1234"><tr><td>x_{init}</td><td>Δx</td><td>y_{max}</td></tr></table> 其中 $y_{max} = \max\{y_i, y_{i+1}\}$ ， $\Delta x = \begin{cases} \frac{x_i - x_{i+1}}{y_i - y_{i+1}}, & y_i \neq y_{i+1} \\ 1, & y_i = y_{i+1} \end{cases}$ $x_{init} = \{(x_k, y_k) k = i, i + 1 \text{ 且 } y_k \neq y_{max}\}$ 将边e放在扫描线 y_{min} 的 NET 中。 具体填充步骤（AET 的动态更新）： (1) 初始化活性边表 AET； (2) 对每条扫描线 i，进行： (a) 将NET[i]中的边加入 AET； (b) 排序使按 x 坐标递增顺序排列； (c) 遍历 AET，将交点两两匹配，可能存在相同的 x 坐标，此时考察其所在边的 y_{max} 是否与当前扫描线 i 相等，若存在 <u>一个相等另一个不等</u> 的情况，将两个点视为一个点，如图；			x_{init}	Δx	y_{max}
x_{init}	Δx	y_{max}			



交点合并示意图

- (d) 填充匹配点对所对应的区间;
- (e) 更新 AET, 将 $y_{max} = i$ 的结点从 AET 中删除, 并把 $y_{max} > i$ 的结点 x 值递增 Δx ;

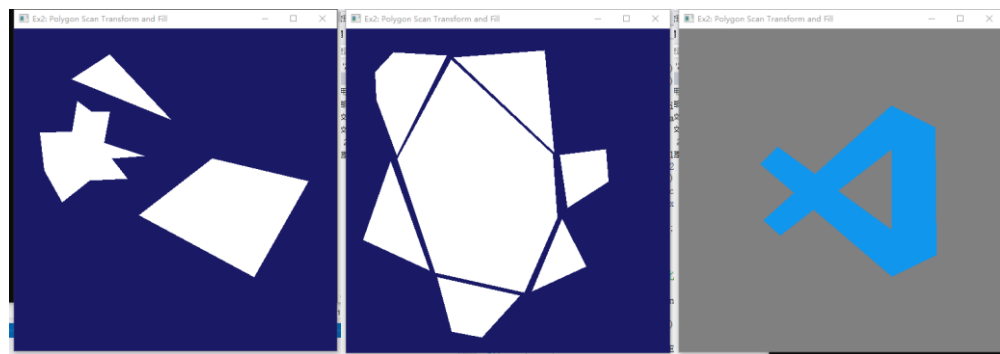
以上。

实验步骤:

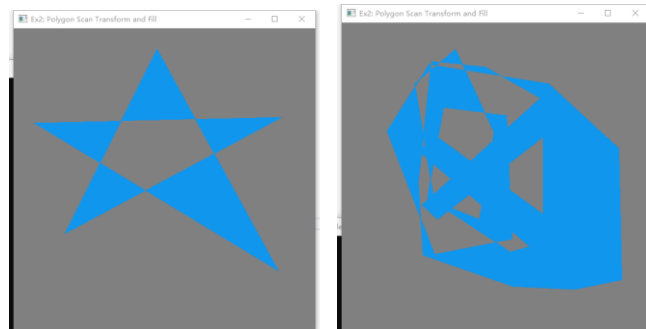
1. 设计交互, 编写鼠标点击事件的回调函数;
2. polygon 扫描转换得到新边表 NET;
3. 自下而上进行扫描, 填充多边形内部;
4. 绘制多种多边形, 进行测试。

实验结果展示及分析:

1. 绘制边不存在自相交情况的多边形:



2. 绘制边存在自相交情况的多边形:



3. Todo



之后想尝试通过图形学的知识在像素层次还原 vsc 的图标, 可以看到它包含了反走样、消隐、曲线等后续课程会学到的内容, 目前通过多边形绘制了大致形状。

实验中存在的问题及解决:

调试时遇到一个 bug, “若隐若现” (跟绘制的图形有关)。苦苦找了半天, 最后发现是按 x 坐标排序那个步骤出问题, 一开始为了效率, 选择有序插入, 但忽略了交点处的特殊情况, 一开始相等向上扫描会出现一大一小, 插入时应同时比较 Δx 。为了容纳边自相交情况, 干脆统一插入后 `sort()`。