

题目：

公司 A 准备开发这样一种设计软件，用户可以向一个平面上添加多个矩形，针对每个矩形，用户可以赋予不同的颜色，如下图所示。在用户操作的时候，软件会后台记录日志。公司需要通过日志分析知道用户编辑完成之后，平面上每个矩形的颜色是什么，请你写一个程序帮他们**逐步实现**这个功能。



程序输入的基本格式为

输入内容	解释
N	//表示接下来会有 N 条操作日志(N<100)
Add normal P1	//一条操作命令
...	//省略 N-1 命令
Normal	//指定输出的格式（此条命令不包含在 N 条操作日志中）

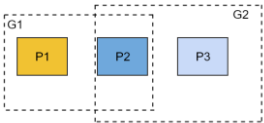
基本需求：用户会依次向平面添加普通的矩形。每个矩形拥有一个编号来标志身份，同时拥有一个颜色属性来标记自身的颜色，颜色用 RGB 的方式表示。在矩形被加入到平面时，会被指定唯一的 polygonid(由 P+数字组成，P1)，矩形默认为黑色，对应的 RGB 表示为(0,0,0)。用户会向平面中指定的矩形填充颜色，如将编号 Pi 的矩形的颜色修改为 (255,0,0)。请你在用户修改完成之后，按照矩形的**编号中的数字部分从小到大的顺序**输出矩形的**编号和颜色**。

衍进需求 1：用户会朝平面添加 3 种矩形，第一种是**普通矩形**，没有特别之处，第二种是**反色矩形**，如果用户为其填充颜色 (X,Y,Z)，那么矩形颜色会被修改成 (255-X,255-Y,255-Z)，第三种是**单红色矩形**，如果用户为其填充颜色 (X,Y,Z)，那么矩形的颜色会被修改为 (X,0,0)。(反色矩形和单红色矩形初始都为黑色)

衍进需求 2：可以选择**按照多表形灰度值从小到大的顺序**输出矩形的编号和颜色，如果灰度值相同，那么按照编号中数字部分从小到大的顺序输出编号和颜色。由 RGB 计算灰度值的公式如下。

$$\text{Gray} = R*0.299 + G*0.587 + B*0.114$$

衍进需求 3：用户觉得一个个修改颜色太过麻烦，用户会选择现将一部分矩形编辑成一个组，并且会为组指定一个 groupid (由 G+数字组成，如 G1)，用户可以直接指定矩形的编号来赋予颜色，或者指定 groupid 来给所有属于这个 Group 的矩形来赋予颜色。(同一个矩形可能会属于多个组，如下图)



对于上述的每个需求：实现基本需求，可以通过 **20%** 的测试用例；实现基本需求+衍

进需求 1，可以通过 50% 的测试用例；在完成前两个需求的基础上，实现衍进需求 2，可以通过 70% 的测试用例；实现全部 4 个需求可以通过全部测试用例。

日志中操作命令说明：

操作命令格式	
Add normal/single/reverse ploygonid	添加一个普通/单色/反色矩形，将其编号为 ploygonid
Group N index1 index2 ... indexN groupid	N 代表后面编号的个数 ，将编号为 index1 到 indexN 的矩形编为一个组，小组的编号为 groupid
Set index red green blue	将编号为 index 的对象颜色修改为 (red,green,blue)
Normal/Gray	按照矩形编号数字部分/灰度值顺序输出

输入输出样例：

输入内容	解释
3 Add normal P2 Set P2 255 0 0 Add normal P1 Normal	//接下来有 3 条操作日志 //添加一个普通矩形，编号为 P2 //将 P2 的颜色修改为(255,0,0) //添加一个普通矩形，编号为 P1 //按照编号数字部分大小顺序输出
输出	
P1 0 0 0 P2 255 0 0	

输入内容	解释
7 Add single P2 Set P2 255 0 0 Add normal P1 Group 2 P2 P1 G1 Add reverse P3 Group 2 P1 P3 G2 Set G2 0 255 0 Gray	//接下来有 7 条操作日志 //添加一个单色矩形，编号为 P2 //将 P2 的颜色修改为(255,0,0) //添加一个普通矩形，编号为 P1 //将 P2 与 P1 归为一个组，组的编号为 G1 //添加反色矩形，编号为 P3 //将 P1 和 P3 编为一组，组的编号为 G2 //将 G2 中所有矩形的颜色修改为 (0,255,0) //按照灰度值大小顺序输出
输出	
P2 255 0 0 P3 255 0 255 P1 0 255 0	

说明：

请大家按照需求顺序逐步实现！

本次机考允许使用 STL 库