

OOP Final Project

概要

Pj是之前lab系列的增量，需完成的内容为：

1. rect指令
2. 交互式
3. undo/redo
4. 统计信息的增强
5. 一个简短的分析报告

增加一个矩形指令

指令`rect(x1, y1, width, height)`为一个左下角为(x1,y1)，宽度(x轴方向)为width，高度(y轴方向)为height的矩形。注意无论在什么样的坐标系下，当width和height为正数时，矩形的另一个顶点位于屏幕的右上角。

交互模式执行

如果运行draw命令时不带文件名作为参数，则进入交互模式，在交互模式下可以输入指令，回车后直接执行并输出结果。

在交互模式下除了已有的指令，增加下面两条指令：

1. 装入在外部文件中定义的宏指令

注意：宏指令不能在交互模式中定义，必须从外部装入。

例如，外部文件predef.txt中的内容为：

```
//带一个下划线的时间
#startTimeView(20,50){
    color(128)
    line(0,0,49,0)
    text(0,3,"08:55")
}
//另一个带下划线的时间
#finishTimeView(20,50){
    color(80)
    line(0,0,49,0)
    text(0,3,"13:55")
}
```

使用`using("predef.txt")`会装入predef.txt中的定义的宏指令，装入后才可以使用`!startTimeView(0,0)`调用。

注意，假设using的文件中只有宏指令的定义。

2. show

在任何时候运行`show`都会打印出当前绘制的图像。

支持undo/redo

可以在交互模式中使用undo/redo。能够被undo/redo 的命令为下列在交互模式录入的指令：

- line
- text
- rect
- !view
- color

比如下面的指令序列：

```
color(1)
line(1, 2, 48, 47)
color(0)
text(5, 45, "Bounday") //0
undo          //1
undo          //2
text(5, 45, "Bounday") //3
undo          //4
redo          //5
```

下面是执行到 `//0` 的位置后，继续往下执行的效果：

- undo //1

```
color(1)
line(1, 2, 48, 47)
color(0)
```

- undo //2

```
color(1)
line(1, 2, 48, 47)
```

- text(5, 45, "Bounday") //3

```
color(1)
line(1, 2, 48, 47)
text(5, 45, "Bounday")
```

- undo //4

```
color(1)
line(1, 2, 48, 47)
```

- redo //5

```
color(1)
line(1, 2, 48, 47)
text(5, 45, "Bounday")
```

为了能够undo，每个能够undo/redo的涉及到屏幕绘制的指令可以在绘制前(1)保存整个屏幕区域，也可以(2)只保存该指令会覆盖的矩形区域(有更高效率的算法，但我们这里可以不考虑)。选择方式(2)的作为Bouns给分。

如果当前状态处于无法undo或redo时，接收到undo/redo命令，可以忽略。

undo/redo后需要打印出结果

统计信息

1. 运行stat会打印当前的统计信息，内容包含下面三行：

- 总执行指令数量：X条
- 输出直线的总长度：X（截断部分也计入）
- 所有输出字符串：aaa,BB,cccc（用“,”分割，不需考虑字符串内部的“,”的转义，截断部分也计入）

其中，输出直线的总长度包括矩形边长的累计。

注意：统计信息中对于undo的支持会很有挑战，特别是对宏指令执行的undo，这部分作为Bouns

2. 我们认为输出文本时，如果用的颜色是当前设备所支持的最亮的颜色(1或255)，这段文本就是包含了非常重要的内容，需要把这段文本单独记录到当前目录下的一个important.txt的文本文件中。

重点文字记录**不需要**支持undo

分析报告

我们已经给出了lab系列的一个**参考设计**，使用了相应的设计模式。但是**Pj实现**并不要求大家一定按照这个设计来，因为这个设计也有不足之处，或者大家修改也来不及。要求大家将自己的**Pj实现**与**参考设计**做一个比较，报告内容主要包括：

1. 对于**参考设计**和**Pj实现**相同的部分，可以分析**参考设计**的优点和不足之处。
2. 对于**参考设计**和**Pj实现**不同的部分，可以对比分析各自的优点和不足之处。
3. Pj的增量部分是否有利用了某些设计决策的优点加快了开发、提高了质量。

字数不是越多越好，没有强制要求，建议2000字以内。

评分标准

项目	分值	说明
rect	15%	
using	10%	配合调用
其他原有指令	10%	
undo/redo	20%	
统计信息1	5%	
统计信息2	5%	
代码结构	10%	模块化，依赖关系的设计
分析	25%	提交一份pdf文档
undo性能	3%	bouns
统计信息1的undo	8%	bouns