公共文档

1. 用户命令的定义
   1. 整体的格式

每个用户的命令文件：：=

头部

{

命令行

}

以.txt后缀保存文件，调用程序可以读取。

* 1. 单个命令格式

命令式样：name('arg\_name':arg1\_value,’arg\_name’:arg2\_value,…)

由四部分组成：

命令名：name

命令作用域：()

参数名：'arg\_name'

参数作用域：:arg1\_value,（冒号起，逗号终）

1. 命令行模块

实现功能：将用户输入命令转换为命令序列，从输入命令中读取到有关图形的信息

## 命令式样

`name('arg\_name':arg,...)`\

说明：\

1.`name`：命令名，如`line`、`circ`、`rect`、`poly`以及一些运算命令等等\

2.`'arg\_name'`：参数名，如`point`、`vector`、`center`、`radius`、`diameter`、`angle`、`length`以及其他的运算命令。\

3.`arg`：参数值，可以是标量也可以是矢量，矢量用括号括起来。\

4.如果不带有参数名，则需要全都不带参数名，否则语法定义为非法。\

5.如果使用带有参数名的命令，则可以不考虑参数的位置，只要在函数作用的括号范围内均视为有效。\

|  |  |
| --- | --- |
| 两点绘直线 | line('point':( x1,y1 ),'point':( x2,y2 )) |
| 点、向量、模长绘直线 | line('point':( x1,y1 ),'vector':( x2,y2 ),'length':len ) |
| 点、向量绘直线 | line('point':(x1,y1),'vector':(x2,y2)) |
| 默认直线 | line((x1,y1),(x2,y2)) |
| 半径、圆心绘圆 | circ('center':( x1,y1 ),'radius':r) |
| 直径、圆心绘圆 | circ('center':( x1,y1 ),'diameter':d) |
| 三点绘圆 | circ('point':(x1,y1),'point':( x2,y2 ),'point':( x3,y3 )) |
| 默认圆 | circ((x1,y1),r) |
| 两点绘矩形 | rect('point':( x1,y1 ),'point':( x2,y2 )) |
| 顶点宽高绘矩形 | rect('point':( x1,y1 ),'height':h,'width':w) |
| 中心宽高绘矩形 | rect('center':(x1,y1),'height':h,'width':w) |
| 中心半径绘正方形 | rect('center':(x1,y1),'radius':r) |
| 默认矩形 | rect((x1,y1),h,w) |
| 默认正方形 | rect((x1,y1),r) |
| 中心、半径、边数绘正多边形 | poly('center':(x1,y1),'radius':r,'sides':s) |
| 中心、直径、边数绘正多边形 | poly('center':( x1,y1 ),'diameter':d,'sides':s) |
| 默认正多边形 | poly((x1,y1),r,s) |

## 直线命令

### 带有两个用户坐标点参数的命令

重载1：`line('point':( x1,y1 ),'point':( x2,y2 ))`

### 带有一个用户坐标点，一个用户向量和一个用户坐标系下的模长的命令

重载2：`line('point':( x1,y1 ),'vector':( x2,y2 ),'length':len )`\

说明：这里的向量可以不是单位向量

### 带有一个用户坐标点，一个用户向量的命令，向量模长即为直线长度

重载3：`line('point':(x1,y1),'vector':(x2,y2))`

### 默认情况，即两个用户坐标点的命令

重载4：`line((x1,y1),(x2,y2))//default`

-----------

## 圆命令

### 带有一个用户坐标点（作圆心），一个用户坐标系下的半径的命令

重载1：`circ('center':( x1,y1 ),'radius':r)`

### 带有一个用户坐标点，一个用户坐标系下的直径的命令

重载2：`circ('center':( x1,y1 ),'diameter':d)`

### 带有三个用户坐标点的命令

重载3：`circ('point':(x1,y1),'point':( x2,y2 ),'point':( x3,y3 ))`

### 默认情况，即一个用户坐标点，一个用户坐标系下的半径的命令

重载4：`circ((x1,y1),r)//default`

-------------

## 矩形命令

### 带有两个用户坐标点的命令

重载1：`rect('point':( x1,y1 ),'point':( x2,y2 ))`

### 带有一个用户坐标点，一个用户坐标系下的高度和一个用户坐标系下的宽度的命令

重载2：`rect('point':( x1,y1 ),'height':h,'width':w)`

### 带有一个用户坐标点（作中心），一个用户坐标系下的高度和一个用户坐标系下的宽度的命令

重载3：`rect('center':(x1,y1),'height':h,'width':w)`

### 带有一个用户坐标点，一个用户坐标系下的半径的命令，画出来的将是一个正方形

重载4：`rect('center':(x1,y1),'radius':r)`

### 默认情况，带有一个用户坐标点，一个用户坐标系下的高度和一个用户坐标系下的宽度的命令

重载5：`rect((x1,y1),h,w)//default`

### 默认情况，带有一个用户坐标点，一个用户坐标系下的半径的命令，画出来的将是一个正方形

重载6：`rect((x1,y1),r)//default`

-------------

## 正多边形命令

### 带有一个用户坐标点（作中心），一个用户坐标系下的半径和一个边数的命令，默认多边形的第一个顶点位于中心的水平连线上

重载1：`poly('center':(x1,y1),'radius':r,'sides':s)`

### 带有一个用户坐标点（作中心），一个用户坐标系下的直径和一个边数的命令

重载2：`poly('center':( x1,y1 ),'diameter':d,'sides':s)`

### 默认情况，即带有一个用户坐标点（作中心），一个用户坐标系下的半径和一个边数的命令

重载3：`poly((x1,y1),r,s)//default`

--------------

## 这个组合命令过于复杂，实现不了就不实现了

## 组合形状命令

### 命令格式如下：用花括号将所有基本图形的命令合在一起，从上到下用换行隔开，组合函数里面可以输入参数，参数必须有且仅有一个

```

group(''){

}

```

### 求并集，即求所有图像叠放后占据了最大面积的形状轮廓

重载1：`group('or')`

### 求交集，即从上到下，依次求两个图像的公共轮廓

重载2：`group('and')`

### 求差集，即从上到下，依次从前一个图像中除去与后一个图像的公共轮廓

重载3：`group('minus')`

----------

## 这个变换命令也比较复杂，实现不了就也不实现了

## 变换命令

### 命令格式如下：用花括号将所有基本图形的命令合在一起，从上到下用换行隔开，变换函数里面可以输入参数

```

trans(''){

}

```

### 平移，带有一个用户向量的命令，默认平移中心为质心

重载1：`trans('move','vector':(x,y))`

### 旋转，带有一个用户角度的命令，默认旋转中心为质心

重载2：`trans('rotate','angle':a)`

### 伸缩，带有一个伸缩因子的命令，默认伸缩中心为质心

重载3：`trans('zoom','scale':s)`

1. 软件整体组成

本软件主要包含四个部分，分别是命令行解码模块，数学几何运算模块，图形用户界面模块，坐标系管理模块和通用模块。

数据流按以下进行：用户输入UI-json-txtcmd命令序列-->stdcmd标准化数据序列-->graph形状序列-->stdcmd标准化数据序列-->json-->UI画面显示

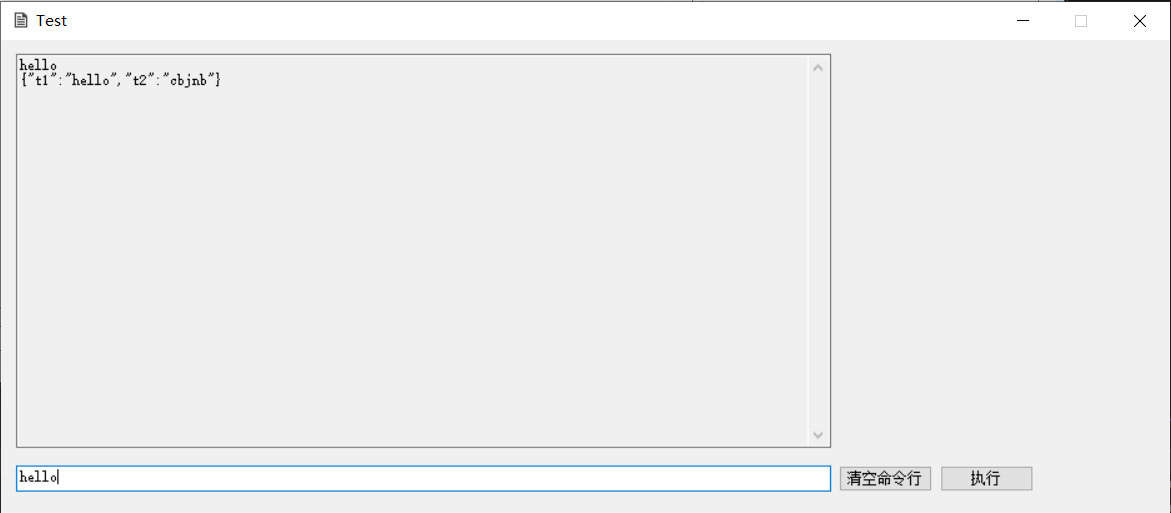
* 1. 命令行解码模块

实现功能：将用户输入的命令提取数据和命令信息，转化为命令序列，然后在对应的标准化数据序列中进行对应的操作，如新建形状、删除形状、形状变换等。

* 1. 数学几何运算模块

实现功能：保证整个软件过程中的数学代数和几何运算，提供诸如坐标平移、旋转、求交并补等运算。

* 1. 图形用户界面



提供与用户交互的界面，同时绘制出用户命令对应的结果。

* 1. 标准化数据序列

1. 分工与时间计划

分工：

命令序列解码：陈博非

基本形状：张国勇

通用线性表：牛泽坤

数学几何运算模块：梁耀星

标准化数据序列：李翔

图形用户界面：曹北健