北京邮电大学

计算机学院（示范性软件学院）

实验报告

课程名称： C语言程序设计

项目名称： 画板程序

项目完成人：

姓名： 杨宇航 学号： 2020522059

指导教师： 杨 谈

日 期： 2020 年 12 月 25 日

# 实验目的

1.理解并掌握 while 语句和 for 语句的基本使用方法。

2.了解 break 和 continue 语句的作用。

3.理解并掌握 if 语句和 switch 语句的基本使用方法。

4.了解函数作为简化程序结构的工具的重要性。

5.了解函数调用的概念以及为什么要在函数调用中提供参数。

6.了解什么是函数原型以及如何编写函数原型。

7.理解在函数间传递信息的机制。

8.能够创建新的函数，并实现调用。

9.学会如何将模块化的程序分文件进行管理。

10.掌握如何定义结构体和声明结构体变量 。

11.掌握如何定义数组及数组的使用 。

12.掌握字符数组和结构体数组的使用。

13.掌握指针的基本操作。

14.理解指针和数组间的关系。

15.进一步理解文件的概念以及文本文件如何实现永久存储。

16.能够创建、读取、写入和更新文件。

# 实验内容

1、实现画板程序功能：询问用户是否读取画板文件。

2、实现画板程序功能：用户根据提示菜单选择要进行的操作。

3、实现画板基本控制逻辑。

4.将实验一中实现的两个模块修改为两个函数，通过一个main 函数来调用两个函数分别实现“读取文件”、“选择要进行的操作”这两个功能。

5.将所有模块分文件存储。

6.绘制更复杂的图形。

7.利用数组和字符串的知识完成更复杂的任务。

8.利用结构体来存储各个图形，来实现撤销等功能。

9.基于指针对结构体数组进行操作。

10.将结构体数组中存储的图形信息存到文件中。

11.将画板中的图形数据从之前保存的文件中导入，并可以直接重绘这些导入的图形。

# 实验环境

Visual Studio集成开发环境和EGE图形库

# 附录

## 问题分析

### 目标

基本功能：

1.根据用户操作绘制图形（含容错）

2.保存所有图形信息至“画板文件”

3.从“画板文件”中读取信息并重绘图形

拓展功能：

1.撤销操作

2.鼠标操作

### 功能

#### 功能一：绘画功能

此画板程序由于时间原因并未加入过多绘画功能，目前为止已有的可绘画图形为：矩形、圆形、实线和多边形。

矩形：鼠标第一次点击的点为矩形左上角，第二次为矩形右下角。

圆形：第一次点击的点为圆的圆心，第二次点击的点与第一次的直线距离为半径。

线：从第一次点击的点到第二次点击的点画一条直线。

多边形：由键盘输入各个点坐标。

#### 功能二：撤销功能

此画板程序运用了结构体数组来记录每个图形的相应信息，选中撤销功能后，程序会把最后一次记录的图形结构体中的数据全部清零，清空画板画图区域后，依据结构体数组中所记录的数据重新再画一次，以此来达到撤销的目的。

#### 功能三：写入、读取文件功能

选中写入、读取文件功能后会弹出一个对话框，再输入文件名后，本储存在结构体中的数据会格式化输入到所写的文件名里

### 健壮性

#### 异常输入处理

1. 菜单异常输入处理：程序的作者将每个菜单都用一个循环来运行其中的程序，如果输入了错误的选项则会显示输出异常并显示重新输入或按任意键继续来从新运行这段代码或跳到下一段代码。
2. 对话款异常输入处理：①输入的文件不存在：程序会对输入的文件名在相应的地址检索，如果不存在该文件，则会显示该文件不存在，并跳过后面一段区域的代码。②输入的多边形坐标不合法：如果输入的多边形坐标内含非数字或逗号部分或输入的逗号为偶数（最后一个坐标只有x轴的坐标而没有y轴的坐标），则会显示输入的坐标有误。

## 设计方案

（在此处，用文字描述整体的设计，包括，分成多少个模块，多少个文件，多少个函数，每个模块、函数、文件的大致功能； 此外，还需要包含一个程序整体的流程图，反映各个模块的执行次序和关系；但是避免绘制细节的流程）

此项目总共分为3个模块，分别为：画图前的预处理模块、画图模块、结束模块。功能如各模块名。

文件分为5个，本别是：画图前的预处理文件、画图文件、结束文件、函数文件、程序主干文件以及readme。功能如各文件名。

函数：

void zone(void)边框

char chooseColor(void)选择颜色

void rechooseColor(char color)选择颜色

char chooseFillColor(void)选择填充颜色

void rechooseFillColor(char fillColor) 选择填充颜色

void clearMenu(void)清理菜单

void clearBoard(void)清理画图区域

void menu(int readResult)菜单分区

int readFile(void)读取文件

void writeToFile(shape\* graphArr)写入文件

initgraph()初始化画板

setcaption()设置窗口标题

setbkcolor()设置背景颜色

setfont()设置字体

outtextxy()输出字符串

getch()读入键盘输入字符

getmouse()获取鼠标信息

inputbox\_getline()弹出对话框

sqrt()开根号

cleardevice()清理画板

delay\_ms()延迟

closegraph()关闭画板

line()画线

rectangle()画矩形

circle()画圆

fillpoly()画多边形

setfillcolor()设置填充颜色

## 实验结果

打开画板显示最初的菜单（读取/不读取文件模块）。选中读取文件，弹出对话框，输入文件名。文件存在，画出所存文件中的数据，进入画图模块；文件不存在，跳过余下步骤，进入画图模块。选中不读取文件，进入画图模块。输入错误，默认不读取文件，进入画图模块。进入画图模块，选中画图选项，先选择想画的图形，再选择颜色，之后鼠标点击或键盘输入进行绘画；选中清除屏幕选项（重画选项），将画图区域清空；选中撤销选项，撤销上一步画的图形；选中保存选项，弹出对话框输入文件名，保存数据；选中退出选项，退出画板程序。

## 调试心得

某些警告或者处理（在自己不会的情况下），最直接的处理就是问别人、在群里问，当然自立是一项十分好的品德，我们可以百度，强烈推荐CSDN基本上大多数的bug在CSDN上都有解决办法。例如在switch的case里面是不能直接赋值的，这中问题完全可以去CSDN上周到解决方法，只要在case里面加上一个大括号就可以解决。在写代码前，思路及逻辑十分重要，在写代码前一定一定一定要先画流程图，并且对着流程图先进行优化，之后再去写代码，这会省下很多没必要花的时间。