

# 智能系统的设计与仿真

## 作业二

姓名：石若川 学号：2111381 专业：智能科学与技术

### 1 实验内容

创建一个单文档应用程序，在菜单栏添加“五角星”功能键，包含“黑色”、“红色”、“绿色”、“蓝色”及“闪烁”，四个功能。在“闪烁”功能下，添加“开始”和“停止”功能。连续鼠标左键单击五个点，连接五个点绘制五角星，并显示五个点的坐标。通过“黑色”、“红色”、“绿色”、“蓝色”可以调整五角星线条颜色。通过点击“闪烁”功能中的“开始”，可以实现每 500ms 闪现一次，且每次闪现颜色不同。通过点击“停止”，可以实现停止闪烁。对“黑色”、“红色”、“绿色”、“蓝色”，“开始”，“停止”，分别添加快捷方式“ALT+D”、“ALT+R”、“ALT+Y”、“ALT+B”、“ALT+K”“ALT+T”。通过点击保存，实现对五角星的位置和颜色的保存。可以打开保存的文件，可以查看五角星的位置和颜色。

### 2 代码分析

#### 2.1 五角星的绘制及坐标显示

在 View 类中添加 OnLButtonDown 函数以响应鼠标左键点击的消息并保存点击的点坐标。

```
1 void CHomework2View::OnLButtonDown(UINT nFlags, CPoint point)
2 {
3     // TODO: 在此添加消息处理程序代码和/或调用默认值
4     CHomework2Doc* pDoc = GetDocument();
5     CString str;
6     str.Format(_T("鼠标单击位置: (%d,%d)"), point.x, point.y);
7     pDoc->Set(point, str);
8
9     m_point.push_back(point);
10
11     Invalidate();
12
13     CView::OnLButtonDown(nFlags, point);
14 }
```

在 View 类中添加如下代码以绘制五角星并显示点坐标。

```
1 void CHomework2View::OnDraw(CDC* pDC)
2 {
3     CHomework2Doc* pDoc = GetDocument();
4     ASSERT_VALID(pDoc);
```

```
5  if (!pDoc)
6      return;
7
8  // TODO: 在此处为本机数据添加绘制代码
9
10 if (m_point.size() >= 5) {
11     CPen pen(PS_SOLID, 2, m_nColors[pDoc->GetColor()]); // 实线, 宽度为2, 给定颜色
12     CPen* pOldPen = pDC->SelectObject(&pen);
13
14     // 循环绘制五个点与下一个点的连线
15     for (int i = 0; i < 5; ++i) {
16         pDC->MoveTo(m_point[0].x, m_point[0].y);
17         pDC->SetTextColor(m_nColors[pDoc->GetColor()]);
18
19         CString str;
20         str.Format(_T("鼠标单击位置: (%d,%d)"), m_point[i].x, m_point[i].y);
21         pDC->TextOut(m_point[i].x, m_point[i].y, str);
22     }
23     pDC->MoveTo(m_point[4].x, m_point[4].y);
24     pDC->LineTo(m_point[1].x, m_point[1].y);
25
26     pDC->MoveTo(m_point[1].x, m_point[1].y);
27     pDC->LineTo(m_point[3].x, m_point[3].y);
28
29     pDC->MoveTo(m_point[3].x, m_point[3].y);
30     pDC->LineTo(m_point[0].x, m_point[0].y);
31
32     pDC->MoveTo(m_point[0].x, m_point[0].y);
33     pDC->LineTo(m_point[2].x, m_point[2].y);
34
35     pDC->MoveTo(m_point[2].x, m_point[2].y);
36     pDC->LineTo(m_point[4].x, m_point[4].y);
37
38     // 恢复原来的画笔和画刷
39     pDC->SelectObject(pOldPen);
40     //pDC->SelectObject(pOldBrush)
41 }
42 }
```

## 2.2 菜单栏的添加和颜色功能的实现

在 Resource.h 中设定对应功能的 ID 值, 如下所示。

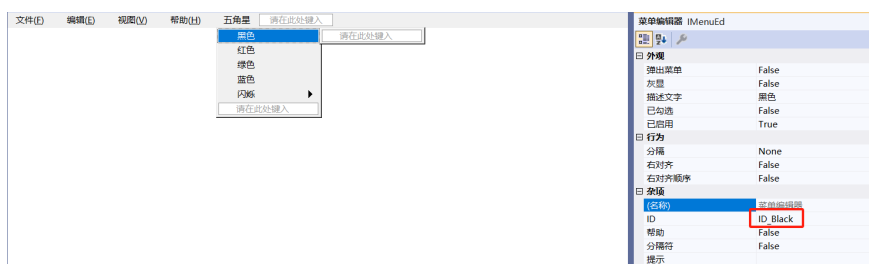


图 2.1: 菜单设置

```

1 #define ID_Black          32778
2 #define ID_Red            32779
3 #define ID_Green         32780
4 #define ID_Blue          32781
5 #define ID_StartFlash    32782
6 #define ID_StopFlash     32783

```

在资源视图-Menu-IDR\_MAINFRAME 中添加所需要的菜单资源。设置一个名为“五角星”的功能键，并包括“黑色”、“红色”、“绿色”、“蓝色”及“闪烁”四个功能。在“闪烁”功能下，添加“开始”和“停止”功能。在 Doc 类中，添加如下代码以对改变颜色的消息进行相应。

```

1 void CHomework2Doc::OnBlack()
2 {
3     // TODO: 在此添加命令处理程序代码
4     m_nColorIndex = 0;
5     UpdateAllViews(NULL);
6 }
7 void CHomework2Doc::OnRed()
8 {
9     // TODO: 在此添加命令处理程序代码
10    m_nColorIndex = 1;
11    UpdateAllViews(NULL);
12 }
13 void CHomework2Doc::OnGreen()
14 {
15    // TODO: 在此添加命令处理程序代码
16    m_nColorIndex = 2;
17    UpdateAllViews(NULL);
18 }
19 void CHomework2Doc::OnBlue()
20 {
21    // TODO: 在此添加命令处理程序代码
22    m_nColorIndex = 3;

```

```
23 UpdateAllViews(NULL);  
24 }
```

在 View 类中，添加如下代码以控制绘图过程中颜色。

```
1 void CHomework2View::OnUpdateBlack(CCmdUI* pCmdUI)  
2 {  
3     // TODO: 在此添加命令更新用户界面处理程序代码  
4     CHomework2Doc* pDoc = GetDocument();  
5     pCmdUI->SetRadio(pDoc->GetColor() == 0);  
6 }  
7 void CHomework2View::OnUpdateRed(CCmdUI* pCmdUI)  
8 {  
9     // TODO: 在此添加命令更新用户界面处理程序代码  
10    CHomework2Doc* pDoc = GetDocument();  
11    pCmdUI->SetRadio(pDoc->GetColor() == 1);  
12 }  
13 void CHomework2View::OnUpdateGreen(CCmdUI* pCmdUI)  
14 {  
15    // TODO: 在此添加命令更新用户界面处理程序代码  
16    CHomework2Doc* pDoc = GetDocument();  
17    pCmdUI->SetRadio(pDoc->GetColor() == 2);  
18 }  
19 void CHomework2View::OnUpdateBlue(CCmdUI* pCmdUI)  
20 {  
21    // TODO: 在此添加命令更新用户界面处理程序代码  
22    CHomework2Doc* pDoc = GetDocument();  
23    pCmdUI->SetRadio(pDoc->GetColor() == 3);  
24 }
```

## 2.3 闪烁功能的实现

对 ID\_StartFlash 和 ID\_StopFlash 进行相应，分别启用和停止一个定时器，并设定定时器的时间间隔为 500ms。在 View 类中添加 OnTimer 函数以响应 WM\_TIMER 消息，在该函数中从 0 到 3 循环设定颜色索引。

```
1 void CHomework2View::OnStartflash()  
2 {  
3     // TODO: 在此添加命令处理程序代码  
4     SetTimer(1, 500, NULL);  
5 }  
6 void CHomework2View::OnStopflash()  
7 {
```

```
8 // TODO: 在此添加命令处理程序代码
9 KillTimer(1);
10 }
11 void CHomework2View::OnTimer(UINT_PTR nIDEvent)
12 {
13 // TODO: 在此添加消息处理程序代码和/或调用默认值
14 color++;
15 if (color == 4) {
16     color = 0;
17 }
18 CHomework2Doc* pDoc = GetDocument();
19 pDoc->SetColor(color);
20 Invalidate();
21 CView::OnTimer(nIDEvent);
22 }
```

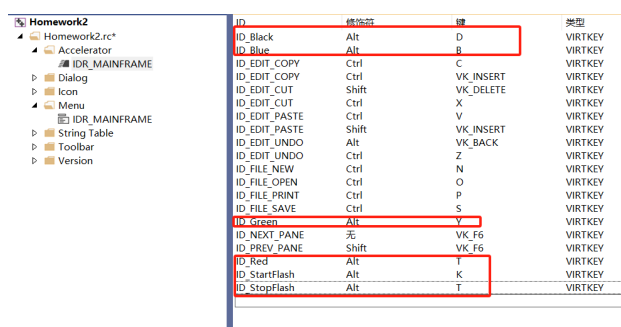
## 2.4 保存文档的实现

对 Doc 类中的 Serialize 函数进行修改，如下代码所示保存每个点的坐标与字符串。

```
1 void CHomework2Doc::Serialize(CArchive& ar)
2 {
3     if (ar.IsStoring())
4     {
5         // TODO: 在此添加存储代码
6         for (int i = 0; i < m_point.size(); i++)
7         {
8             ar << m_point[i] << m_str[i] << m_nColorIndex;
9         }
10    }
11    else
12    {
13        // TODO: 在此添加加载代码
14        for (int i = 0; i < m_point.size(); i++)
15        {
16            ar >> m_point[i] >> m_str[i] >> m_nColorIndex;
17        }
18    }
19 }
```

## 2.5 快捷键的实现

在资源视图-Accelerator-IDR\_MAINFRAME 中进行修改添加黑、红、绿、蓝、开始、停止六个功能所对应的 ID 值，并设置对应的快捷键即可，如图2.2所示。



ID	修饰符	键	类型
ID_Black	Alt	D	VIRTKEY
ID_Blue	Alt	B	VIRTKEY
ID_EDIT_COPY	Ctrl	C	VIRTKEY
ID_EDIT_COPY	Ctrl	VK_INSERT	VIRTKEY
ID_EDIT_CUT	Shift	VK_DELETE	VIRTKEY
ID_EDIT_CUT	Ctrl	X	VIRTKEY
ID_EDIT_PASTE	Ctrl	V	VIRTKEY
ID_EDIT_PASTE	Shift	VK_INSERT	VIRTKEY
ID_EDIT_UNDO	Alt	VK_BACK	VIRTKEY
ID_EDIT_UNDO	Ctrl	Z	VIRTKEY
ID_FILE_NEW	Ctrl	N	VIRTKEY
ID_FILE_OPEN	Ctrl	O	VIRTKEY
ID_FILE_PRINT	Ctrl	P	VIRTKEY
ID_FILE_SAVE	Ctrl	S	VIRTKEY
ID_Green	Alt	G	VIRTKEY
ID_NEXT_PANE	无	VK_F6	VIRTKEY
ID_PREV_PANE	Shift	VK_F6	VIRTKEY
ID_Red	Alt	R	VIRTKEY
ID_StartFlash	Alt	K	VIRTKEY
ID_StopFlash	Alt	T	VIRTKEY

图 2.2: 快捷键设置

## 3 结果展示

在屏幕中点击五个点后，屏幕中会默认显示黑色的五角星以及点坐标（点击超过五个点时，会显示前五个点构成的五角星）。通过点击菜单栏的功能键可以改变五角星的颜色，并使其开始闪烁与停止，如下图所示。

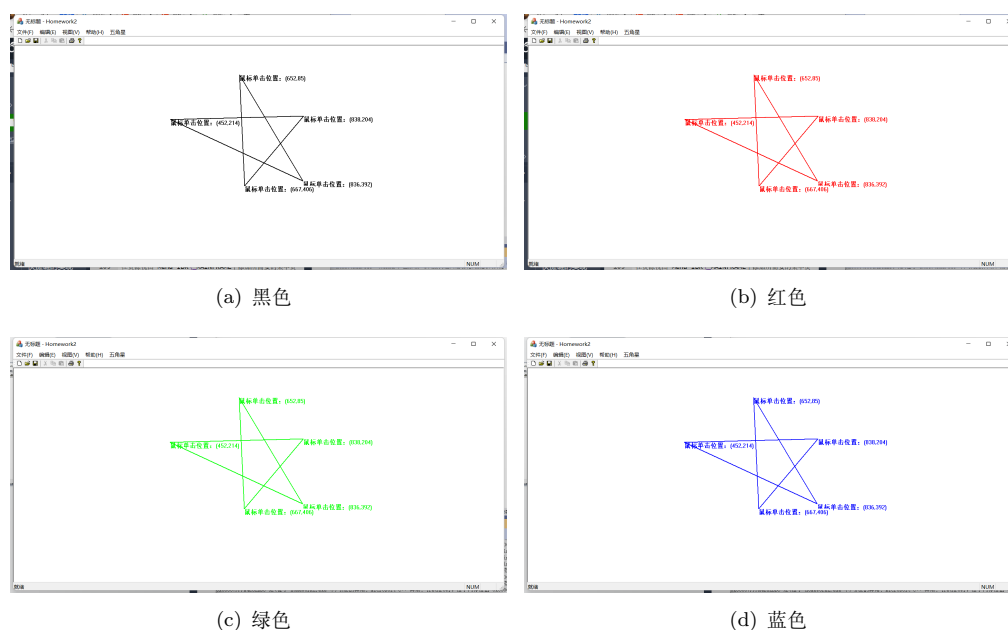


图 3.3: 实验结果

## 4 实验总结

在本次作业中，我实现了五角星的绘制，可以通过菜单资源以及快捷键改变其颜色以及控制闪烁，并可以保存与打开文件。通过本次作业，我学习和掌握了文档与视图的基本用法，并通过菜单资源和快捷键实现指定的功能。