《Visual C++与面向对象程序设计》
实训报告

姓　　名	刘运涛
专业年级	18 生物科学
学　　号	18071001041
授课教师	李庆忠

中国海洋大学工程学院自动化及测控系

2019 年夏季学期

第十二章 图形设备接口和资源

1. 菜单修改折线颜色

题目要求：通过菜单选择改变所画折线的颜色。

程序说明：利用菜单编辑器添加一个“颜色”菜单项。在颜色菜单下添加 3 个子项，分别为**“黄色”、“蓝色”和“绿色”**。将它们的 ID 分别设置为 **ID_COLOR_YELLOW、ID_COLOR_BLUE、ID_COLOR_RED**。同时利用类向导分别为这 3 项添加消息处理函数，并将消息处理函数添加到 CBbbbhgView。

模块组成：折线显示，坐标轴，轴标题及刻度，菜单调节

程序分模块实现：

```
//在 CBbbbhgDoc 类的定义中添加成员变量
class CBbbbhgDoc : public CDocument
{
protected: // create from serialization only
    CBbbbhgDoc();
    DECLARE_DYNCREATE(CBbbbhgDoc)

// Attributes
public:

// Operations
public:

// Overrides
    // ClassWizard generated virtual function overrides
    //{{AFX_VIRTUAL(CBbbbhgDoc)
    public:
    virtual BOOL OnNewDocument();
    virtual void Serialize(CArchive& ar);
    //}}AFX_VIRTUAL
    enum {iPt=10};
    double m_dbXdata[iPt];
    double m_dbYdata[iPt];
    COLORREF m_colorrefColor;
// Implementation
```

```
public:
    virtual ~CBbbbhgDoc();
#ifdef _DEBUG
    virtual void AssertValid() const;
    virtual void Dump(CDumpContext& dc) const;
#endif

protected:

// Generated message map functions
protected:
    //{{AFX_MSG(CBbbbhgDoc)
        // NOTE - the ClassWizard will add and remove member functions here.
        //    DO NOT EDIT what you see in these blocks of generated code !
    //}}AFX_MSG
    DECLARE_MESSAGE_MAP()
};
```

//在 CBbbbhgDoc 的成员函数 OnNewDocument()中初始化数据

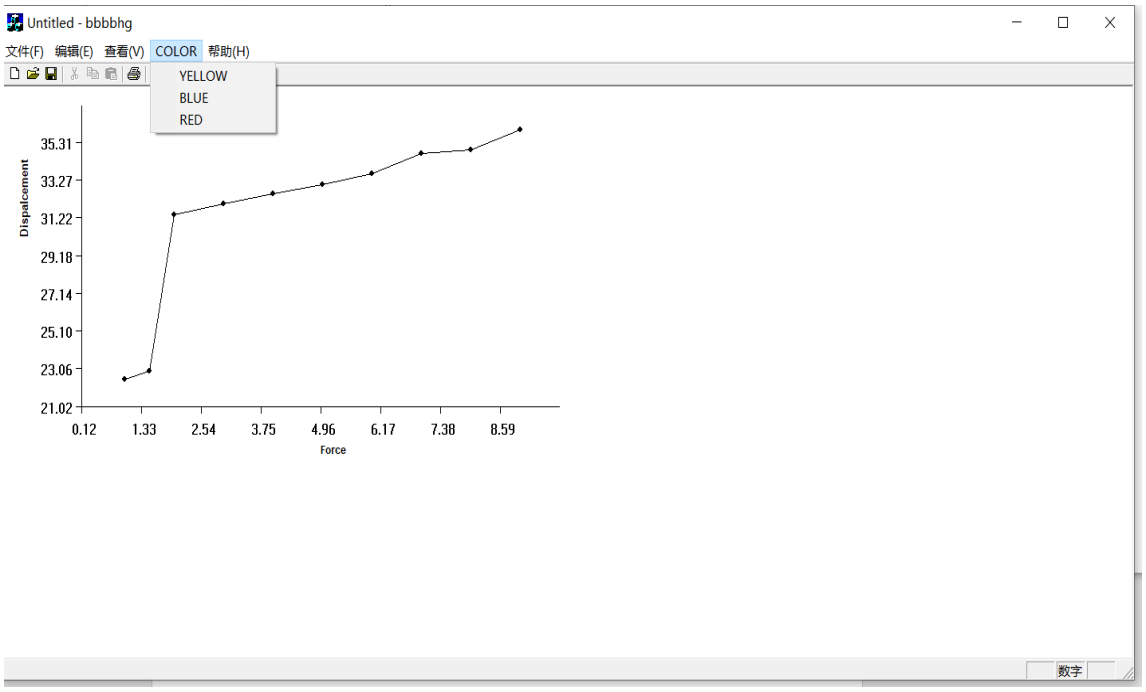
```
BOOL CBbbbhgDoc::OnNewDocument()
{
    if (!CDocument::OnNewDocument())
        return FALSE;

    // TODO: add reinitialization code here
    // (SDI documents will reuse this document)
    m_dbXdata[0]=1;
    m_dbXdata[1]=1.5;
    m_dbXdata[2]=2;
    m_dbXdata[3]=3;
    m_dbXdata[4]=4;
    m_dbXdata[5]=5;
    m_dbXdata[6]=6;
    m_dbXdata[7]=7;
    m_dbXdata[8]=8;
    m_dbXdata[9]=9;
    m_dbYdata[0]=22.5;
    m_dbYdata[1]=22.9;
    m_dbYdata[2]=31.4;
    m_dbYdata[3]=32.0;
    m_dbYdata[4]=32.5;
    m_dbYdata[5]=33;
    m_dbYdata[6]=33.6;
    m_dbYdata[7]=34.7;
    m_dbYdata[8]=34.9;
```

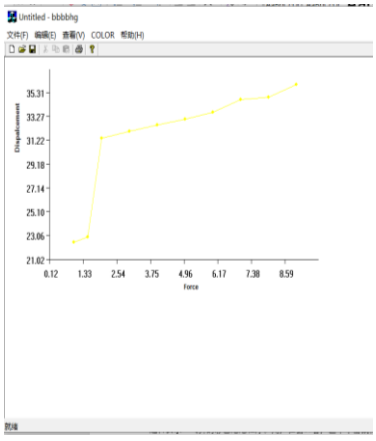
```
    m_dbYdata[9]=36;
    m_colorrefColor=RGB(0,0,0);
    return TRUE;
}
//编辑 3 个新添加的菜单消息处理函数（红，黄，蓝）
void CBbbbhgView::OnColorBlue()
{
    // TODO: Add your command handler code here
    CBbbbhgDoc* pDoc = GetDocument();
    ASSERT_VALID(pDoc);
    pDoc->m_colorrefColor=RGB(0,0,255);
    InvalidateRect(NULL);
}

void CBbbbhgView::OnColorRed()
{
    // TODO: Add your command handler code here
    CBbbbhgDoc* pDoc = GetDocument();
    ASSERT_VALID(pDoc);
    pDoc->m_colorrefColor=RGB(255,0,0);
    InvalidateRect(NULL);
}

void CBbbbhgView::OnColorYellow()
{
    // TODO: Add your command handler code here
    CBbbbhgDoc* pDoc = GetDocument();
    ASSERT_VALID(pDoc);
    pDoc->m_colorrefColor=RGB(255,255,0);
    InvalidateRect(NULL);
}
//编辑 OnDraw()函数显示折线
void CBbbbhgView::OnDraw(CDC* pDC)
{
    CBbbbhgDoc* pDoc = GetDocument();
    ASSERT_VALID(pDoc);
    // TODO: add draw code for native data here
    const int xOrg=100;
    const int yOrg=350;
    const int xMax=700;
    const int yMin=20;
    const int xMin=xOrg;
    const int yMax=yOrg;
```

```
double dbXMin=1;
 double dbXMax=9;
 double dbYMin=22.5;
 double dbYMax=36;

dbXMax=dbXMax+(dbXMax-dbXMin)/pDoc->iPt;
 dbXMin=dbXMin-(dbXMax-dbXMin)/pDoc->iPt;
 dbYMax=dbYMax+(dbYMax-dbYMin)/pDoc->iPt;
 dbYMin=dbYMin-(dbYMax-dbYMin)/pDoc->iPt;

double dbXRatio=(xMax-xOrg)/(dbXMax-dbXMin);
 double dbYRatio=(yOrg-yMin)/(dbYMax-dbYMin);

 CPen penNewPen(PS_SOLID,1,pDoc->m_colorrefColor);
CPen * ppenOldPen=pDC->SelectObject(&penNewPen);
 CBrush brushNewBrush(pDoc->m_colorrefColor);
CBrush * pbrushOldBrush=pDC->SelectObject(&brushNewBrush);
 int x=(int)(dbXRatio*(pDoc->m_dbXdata[0]-dbXMin)+xOrg);
 int y=(int)(yOrg-dbYRatio*(pDoc->m_dbYdata[0]-dbYMin));
 pDC->MoveTo(x,y);
 CRect rectSymbol(x-3,y-3,x+3,y+3);
 pDC->Ellipse(rectSymbol);
 for(int i=1;i<pDoc->iPt;i++)
 {
   x=(int)(dbXRatio*(pDoc->m_dbXdata[i]-dbXMin)+xOrg);
   y=(int)(yOrg-dbYRatio*(pDoc->m_dbYdata[i]-dbYMin));
    rectSymbol.left=x-3;
    rectSymbol.top=y-3;
    rectSymbol.right=x+3;
    rectSymbol.bottom=y+3;
    pDC->Ellipse(rectSymbol);
    pDC->LineTo(x,y);
 }

 pDC->SelectStockObject(BLACK_PEN);
 pDC->SelectStockObject(WHITE_BRUSH);
 pDC->MoveTo(xOrg,yOrg);
 pDC->LineTo(xMax,yOrg);
 pDC->MoveTo(xOrg,yOrg);
 pDC->LineTo(xOrg,yMin);

 const iNumScal=8;
 double dbXInc=(dbXMax-dbXMin)/iNumScal;
 y=yOrg;
```

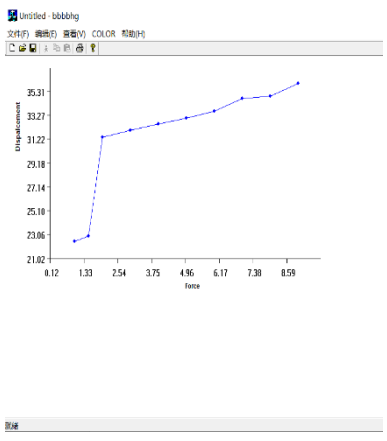
```
    for(i=0;i<iNumScal;++i)
    {
        x=(int)(dbXRatio*i*dbXInc+xOrg);
        pDC->MoveTo(x,y);
        pDC->LineTo(x,y+8);
        CString strBufOut;
        strBufOut.Format("%6.2f",(dbXMin+i*dbXInc));
        pDC->TextOut(x-20,y+15,strBufOut);
    }

    double dbYInc=(dbYMax-dbYMin)/iNumScal;
    x=xOrg;
    for(i=0;i<iNumScal;++i)
    {
        y=(int)(yOrg-dbYRatio*i*dbYInc);
        pDC->MoveTo(x,y);
        pDC->LineTo(x-8,y);
        CString strBufOut;
        strBufOut.Format("%6.2f",(dbYMin+i*dbYInc));
        pDC->TextOut(x-55,y-8,strBufOut);
    }


     CFont fontNewFont;
    fontNewFont.CreateFont(15,0,0,0,0,FALSE,FALSE,0,ANSI_CHARSET,OUT_DEFAULT_PRECIS,CLI
P_DEFAULT_PRECIS,DEFAULT_QUALITY,DEFAULT_PITCH|FF_MODERN,"Arial Bold");
    CFont * pfontOldFont=pDC->SelectObject(&fontNewFont);
    x=(xMax-xOrg)/3*2;
    y=yOrg+40;
    pDC->TextOut(x,y,"Force");
    fontNewFont.DeleteObject();
    fontNewFont.CreateFont(15,0,900,900,0,FALSE,FALSE,0,ANSI_CHARSET,OUT_DEFAULT_PRECIS
,CLIP_DEFAULT_PRECIS,DEFAULT_QUALITY,DEFAULT_PITCH|FF_MODERN,"Arial Bold");
    pDC->SelectObject(&fontNewFont);
    x=xOrg-80;
    y=(yOrg-yMin)/2;
    pDC->TextOut(x,y,"Dispalcement");
    pDC->SelectObject(ppenOldPen);
    pDC->SelectObject(pbrushOldBrush);
    pDC->SelectObject(pfontOldFont);
}
```

运行结果：

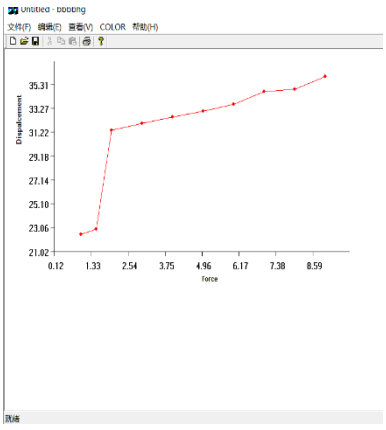

Untitled - bbbbhg
文件(F) 编辑(E) 查看(V) COLOR 帮助(H)
YELLOW
BLUE
RED
Dispalcement
35.31
33.27
31.22
29.18
27.14
25.10
23.06
21.02
0.12 1.33 2.54 3.75 4.96 6.17 7.38 8.59
Force
数字



黄色



蓝色



红色

2. 彩色泡泡

题目要求：飞舞的彩色泡泡程序。用户在窗口客户区中单击鼠标左/右键时即可产生红/蓝色泡泡并向上升起。

程序说明：利用菜单编辑器添加 **“开始”(ID_START)**和**“结束”(ID_END)**菜单项，用·ClassWizard 建立消息处理函数。再利用 ClassWizard 建立鼠标左/右键消息处理函数和定时器消息处理函数，在文档类定义中设置泡泡颜色为红，蓝两色，并初始化，最后使用 OnDraw 绘制泡泡图像。

模块组成：鼠标左/右键消息处理，定时器，定义及初始化（颜色：红/蓝）,图像绘制

程序分模块实现：

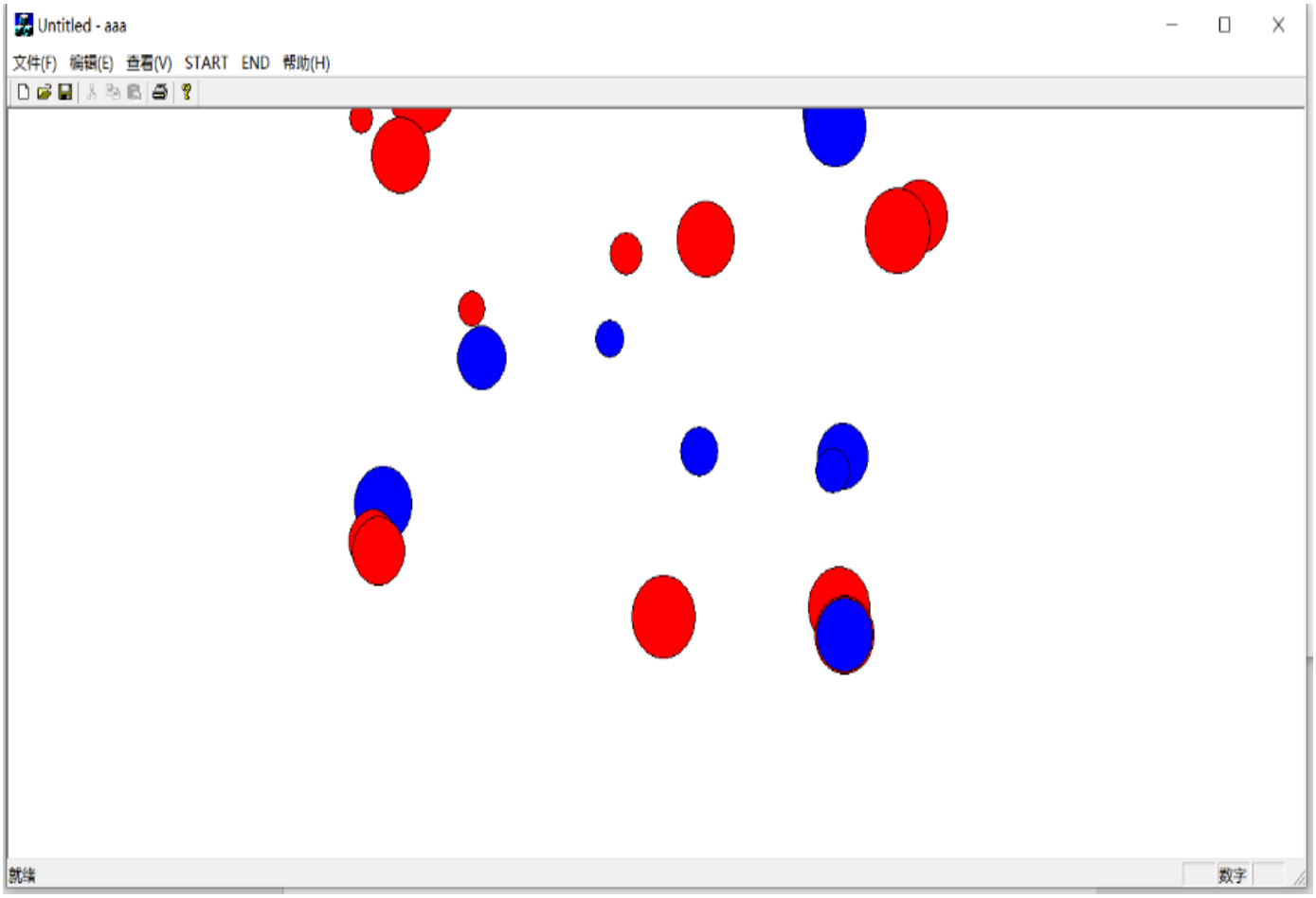
```
//修改文档类的定义，加入宏定义及变量定义
#define MAX_BUBBLE 250
class CAaaDoc : public CDocument
{
protected: // create from serialization only
    CAaaDoc();
    DECLARE_DYNCREATE(CAaaDoc)
// Attributes
public:
    CRect m_rectBubble[MAX_BUBBLE];
    int m_nColor[MAX_BUBBLE];
    int m_nBubbleCount;
    int m_nStart;
// Operations
public:

// Overrides
    // ClassWizard generated virtual function overrides
    //{{AFX_VIRTUAL(CAaaDoc)
    public:
    virtual BOOL OnNewDocument();
    virtual void Serialize(CArchive& ar);
    //}}AFX_VIRTUAL
```

```
// Implementation
public:
    virtual ~CAaaDoc();
#ifdef _DEBUG
    virtual void AssertValid() const;
    virtual void Dump(CDumpContext& dc) const;
#endif

protected:

// Generated message map functions
protected:
    //{{AFX_MSG(CAaaDoc)
        // NOTE - the ClassWizard will add and remove member functions here.
        //    DO NOT EDIT what you see in these blocks of generated code !
    //}}AFX_MSG
    DECLARE_MESSAGE_MAP()
};
//修改文档类的 OnNewDocument()函数，对变量进行初始化；
BOOL CAaaDoc::OnNewDocument()
{
    if (!CDocument::OnNewDocument())
        return FALSE;
    m_nBubbleCount=0;
    m_nStart=0;
    // TODO: add reinitialization code here
    // (SDI documents will reuse this document)
    return TRUE;
}
//修改视图类的 OnDraw 函数
void CAaaView::OnDraw(CDC* pDC)
{
    CAaaDoc* pDoc = GetDocument();
    ASSERT_VALID(pDoc);
    // TODO: add draw code for native data here
    CBrush brushNewRed,brushNewBlue,*pbrushOld;
    brushNewRed.CreateSolidBrush(RGB(255,0,0));
    brushNewBlue.CreateSolidBrush(RGB(0,0,255));
    pbrushOld=pDC->SelectObject(&brushNewRed);
    for(int i=0;i<pDoc->m_nBubbleCount;i++)
        if(pDoc->m_rectBubble[i].bottom>-100)
        {
            if(pDoc->m_nColor[i]==1)
            {
```

```
                pDC->SelectObject(&brushNewRed);
            }
            else
            {
                pDC->SelectObject(&brushNewBlue);
            }
            pDC->Ellipse(pDoc->m_rectBubble[i]);
        }
        pDC->SelectObject(pbrushOld);
}
//建立鼠标左/右键的消息处理函数
void CAaaView::OnLButtonDown(UINT nFlags, CPoint point)
{
    // TODO: Add your message handler code here and/or call default
    CAaaDoc* pDoc = GetDocument();
    ASSERT_VALID(pDoc);
    if(pDoc->m_nStart==1)
        if(pDoc->m_nBubbleCount<MAX_BUBBLE)
        {
            int r=rand()%30+10;
            CRect rect(point.x-r,point.y-r,point.x+r,point.y+r);
            pDoc->m_rectBubble[pDoc->m_nBubbleCount]=rect;
            pDoc->m_nColor[pDoc->m_nBubbleCount]=1;
            pDoc->m_nBubbleCount++;
            InvalidateRect(rect,FALSE);
        }
    CView::OnLButtonDown(nFlags, point);
}

void CAaaView::OnRButtonDown(UINT nFlags, CPoint point)
{
    // TODO: Add your message handler code here and/or call default
    CAaaDoc* pDoc = GetDocument();
    ASSERT_VALID(pDoc);
    if(pDoc->m_nStart==1)
        if(pDoc->m_nBubbleCount<MAX_BUBBLE)
        {
            int r=rand()%30+10;
            CRect rect(point.x-r,point.y-r,point.x+r,point.y+r);
            pDoc->m_rectBubble[pDoc->m_nBubbleCount]=rect;
            pDoc->m_nColor[pDoc->m_nBubbleCount]=2;
            pDoc->m_nBubbleCount++;
            InvalidateRect(rect,FALSE);
        }
```

```
    CView::OnRButtonDown(nFlags, point);
}
//用菜单编辑器加入菜单项“Start”和“End”，并分别建立消息处理函数
void CAaaView::OnStart()
{
    // TODO: Add your command handler code here
    CAaaDoc* pDoc = GetDocument();
    ASSERT_VALID(pDoc);
    pDoc->m_nStart=1;
    SetTimer(1,100,NULL);
}

void CAaaView::OnEnd()
{
    // TODO: Add your command handler code here
    CAaaDoc* pDoc = GetDocument();
    ASSERT_VALID(pDoc);
    pDoc->m_nStart=0;
    KillTimer(1);
}
//建立定时器消息处理函数
void CAaaView::OnTimer(UINT nIDEvent)
{
    // TODO: Add your message handler code here and/or call default
    CAaaDoc* pDoc = GetDocument();
    ASSERT_VALID(pDoc);
    for(int i=0;i<pDoc->m_nBubbleCount;i++)
    {
        if(pDoc->m_rectBubble[i].bottom>0)
        {
            InvalidateRect(pDoc->m_rectBubble[i],TRUE);
            pDoc->m_rectBubble[i].top-=15;
            pDoc->m_rectBubble[i].bottom-=15;
           InvalidateRect(pDoc->m_rectBubble[i],FALSE);
        }
        else
            pDoc->m_rectBubble[i].bottom=-100;
    }
    CView::OnTimer(nIDEvent);
}
```

运行结果：



3. 彩色双折线

题目要求：彩色折线图，用不同颜色和标记画出折线图。

程序说明：为了显示折线颜色，创建一个 **CPen** 的图像对象。为了使画笔能改变颜色，在 CPen 类中先调用成员函数 **DeleteObject()**将它从设备环境中删除，再调用 **CreatePen()**成员函数创建一个新颜色的画笔，再次选入设备中然后再用新的颜色画线。在写 Y 轴的轴标题时，重新创建了一个新的字体对象，该字体对象将字体逆时针旋转 90 度后输出，其他部分与题 2 基本类似。

模块组成：折线颜色，折线显示，坐标轴，轴标题及刻度

程序分模块实现：

```
 //在 OnDraw 中加入折线主体显示部分
void CQqqView::OnDraw(CDC* pDC)
{
    CQqqDoc* pDoc = GetDocument();
    ASSERT_VALID(pDoc);
    // TODO: add draw code for native data here
    const int xOrg=100;
    const int yOrg=350;
    const int xMax=700;
    const int yMin=20;
    const int xMin=xOrg;
    const int yMax=yOrg;

    double dbXMin=1;
    double dbXMax=9;
    double dbYMin1=22.5;
    double dbYMax1=36;
    double dbYMin2=22.5;
    double dbYMax2=39;
    double dbYMin=22.5;
    double dbYMax=39;

    dbXMax=dbXMax+(dbXMax-dbXMin)/pDoc->iPt;
    dbXMin=dbXMin-(dbXMax-dbXMin)/pDoc->iPt;
    dbYMax=dbYMax+(dbYMax-dbYMin)/pDoc->iPt;
    dbYMin=dbYMin-(dbYMax-dbYMin)/pDoc->iPt;
```

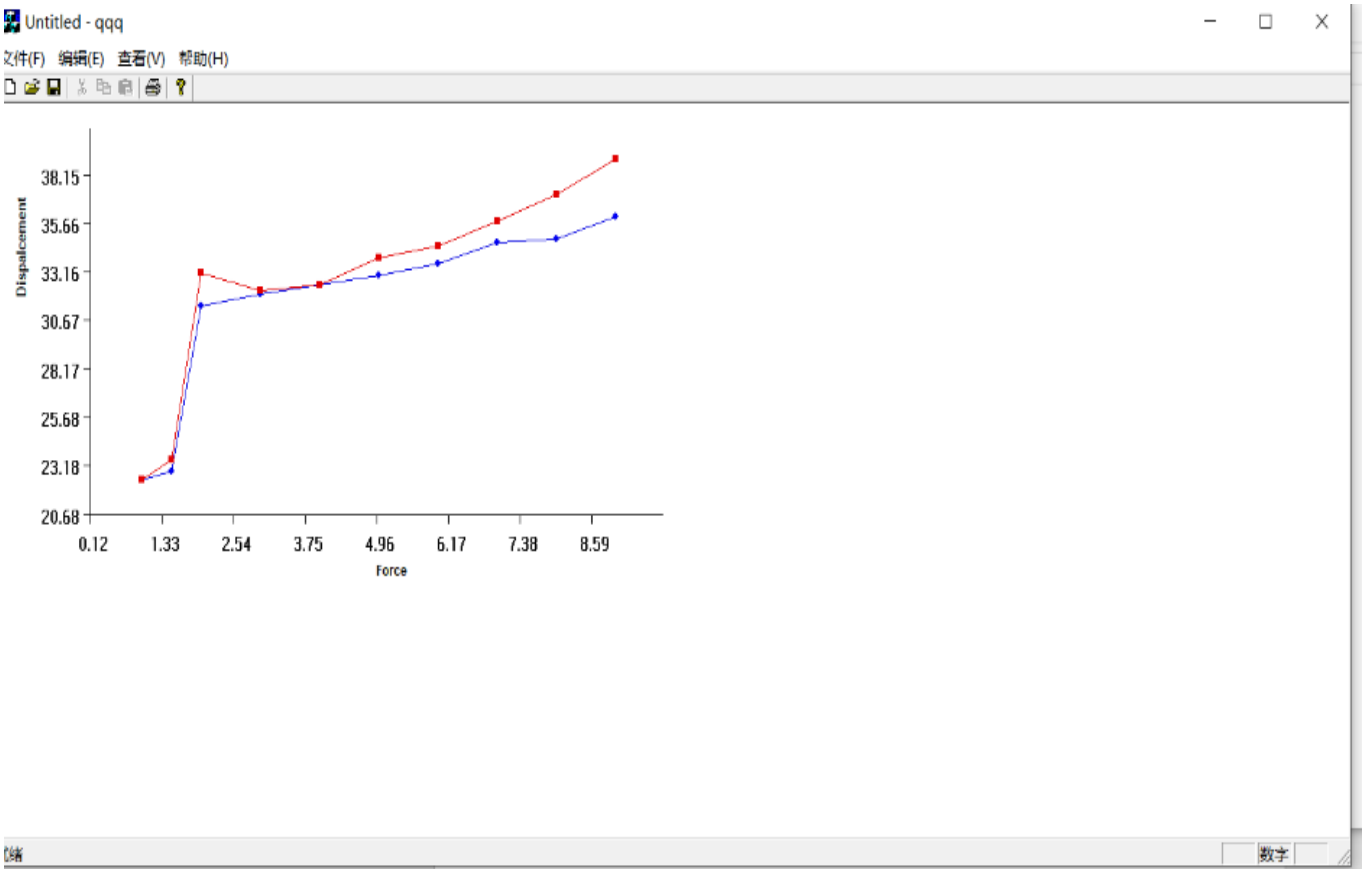
```
double dbXRatio=(xMax-xOrg)/(dbXMax-dbXMin);
double dbYRatio=(yOrg-yMin)/(dbYMax-dbYMin);

CPen penNewPen(PS_SOLID,1,RGB(0,0,225));
CPen * ppenOldPen=pDC->SelectObject(&penNewPen);
CBrush brushNewBrush(RGB(0,0,255));
CBrush * pbrushOldBrush=pDC->SelectObject(&brushNewBrush);
int x=(int)(dbXRatio*(pDoc->m_dbXdata[0]-dbXMin)+xOrg);
int y=(int)(yOrg-dbYRatio*(pDoc->m_dbYdata1[0]-dbYMin));
pDC->MoveTo(x,y);
CRect rectSymbol(x-3,y-3,x+3,y+3);
pDC->Ellipse(rectSymbol);
for(int i=1;i<pDoc->iPt;i++)
{
  x=(int)(dbXRatio*(pDoc->m_dbXdata[i]-dbXMin)+xOrg);
  y=(int)(yOrg-dbYRatio*(pDoc->m_dbYdata1[i]-dbYMin));
   rectSymbol.left=x-3;
   rectSymbol.top=y-3;
   rectSymbol.right=x+3;
   rectSymbol.bottom=y+3;
   pDC->Ellipse(rectSymbol);
   pDC->LineTo(x,y);
}

penNewPen.DeleteObject();
penNewPen.CreatePen(PS_SOLID,1,RGB(225,0,0));
pDC->SelectObject(&penNewPen);
brushNewBrush.DeleteObject();
brushNewBrush.CreateSolidBrush(RGB(225,0,0));
x=(int)(dbXRatio*(pDoc->m_dbXdata[0]-dbXMin)+xOrg);
y=(int)(yOrg-dbYRatio*(pDoc->m_dbYdata2[0]-dbYMin));
pDC->MoveTo(x,y);
rectSymbol.left=x-3;
rectSymbol.top=y-3;
rectSymbol.right=x+3;
rectSymbol.bottom=y+3;
pDC->FillRect(rectSymbol,&brushNewBrush);
for(i=1;i<pDoc->iPt;i++)
{
   x=(int)(dbXRatio*(pDoc->m_dbXdata[i]-dbXMin)+xOrg);
  y=(int)(yOrg-dbYRatio*(pDoc->m_dbYdata2[i]-dbYMin));
  rectSymbol.left=x-3;
   rectSymbol.top=y-3;
```

```
            rectSymbol.right=x+3;
            rectSymbol.bottom=y+3;
            pDC->FillRect(rectSymbol,&brushNewBrush);
            pDC->LineTo(x,y);
        }

        pDC->SelectObject(ppenOldPen);
        pDC->SelectObject(pbrushOldBrush);
        pDC->MoveTo(xOrg,yOrg);
        pDC->LineTo(xMax,yOrg);
        pDC->MoveTo(xOrg,yOrg);
        pDC->LineTo(xOrg,yMin);

       const iNumScal=8;
        double dbXInc=(dbXMax-dbXMin)/iNumScal;
        y=yOrg;
        for(i=0;i<iNumScal;++i)
        {
             x=(int)(dbXRatio*i*dbXInc+xOrg);
             pDC->MoveTo(x,y);
             pDC->LineTo(x,y+8);
             CString strBufOut;
             strBufOut.Format("%6.2f",(dbXMin+i*dbXInc));
             pDC->TextOut(x-20,y+15,strBufOut);
        }
        double dbYInc=(dbYMax-dbYMin)/iNumScal;
        x=xOrg;
        for(i=0;i<iNumScal;++i)
        {
             y=(int)(yOrg-dbYRatio*i*dbYInc);
             pDC->MoveTo(x,y);
             pDC->LineTo(x-8,y);
             CString strBufOut;
             strBufOut.Format("%6.2f",(dbYMin+i*dbYInc));
             pDC->TextOut(x-55,y-8,strBufOut);
        }

       CFont fontNewFont;
        fontNewFont.CreateFont(15,0,0,0,0,FALSE,FALSE,0,ANSI_CHARSET,OUT_DEFAULT_PRECIS,CLI
P_DEFAULT_PRECIS,DEFAULT_QUALITY,DEFAULT_PITCH|FF_MODERN,"Arial Bold");
       CFont * pfontOldFont=pDC->SelectObject(&fontNewFont);
        x=(xMax-xOrg)/3*2;
        y=yOrg+40;
        pDC->TextOut(x,y,"Force");
```

```
    fontNewFont.DeleteObject();
    fontNewFont.CreateFont(15,0,900,900,0,FALSE,FALSE,0,ANSI_CHARSET,OUT_DEFAULT_PRECIS
,CLIP_DEFAULT_PRECIS,DEFAULT_QUALITY,DEFAULT_PITCH|FF_MODERN,"Arial Bold");
    pDC->SelectObject(&fontNewFont);
    x=xOrg-80;
    y=(yOrg-yMin)/2;
    pDC->TextOut(x,y,"Dispalcement");
    pDC->SelectObject(pfontOldFont);
}
//在CQqqDoc的声明中添加成员变量
class CQqqDoc : public CDocument
{
protected: // create from serialization only
    CQqqDoc();
    DECLARE_DYNCREATE(CQqqDoc)
// Attributes
public:
    enum {iPt=10};
    double m_dbXdata[iPt];
    double m_dbYdata1[iPt];
    double m_dbYdata2[iPt];

// Operations
public:
// Overrides
    // ClassWizard generated virtual function overrides
    //{{AFX_VIRTUAL(CQqqDoc)
    public:
    virtual BOOL OnNewDocument();
    virtual void Serialize(CArchive& ar);
    //}}AFX_VIRTUAL

// Implementation
public:
    virtual ~CQqqDoc();
#ifdef _DEBUG
    virtual void AssertValid() const;
    virtual void Dump(CDumpContext& dc) const;
#endif
protected:
// Generated message map functions
protected:
    //{{AFX_MSG(CQqqDoc)
        // NOTE - the ClassWizard will add and remove member functions here.
```

```
        //    DO NOT EDIT what you see in these blocks of generated code !
    //}}AFX_MSG
    DECLARE_MESSAGE_MAP()
};
//在 CQqqDo 类成员函数 OnNewDocument()中初始化成员变量
BOOL CQqqDoc::OnNewDocument()
{
    if (!CDocument::OnNewDocument())
        return FALSE;

    // TODO: add reinitialization code here
    // (SDI documents will reuse this document)
    m_dbXdata[0]=1;
    m_dbXdata[1]=1.5;
    m_dbXdata[2]=2;
    m_dbXdata[3]=3;
    m_dbXdata[4]=4;
    m_dbXdata[5]=5;
    m_dbXdata[6]=6;
    m_dbXdata[7]=7;
    m_dbXdata[8]=8;
    m_dbXdata[9]=9;
    m_dbYdata1[0]=22.5;
    m_dbYdata1[1]=22.9;
    m_dbYdata1[2]=31.4;
    m_dbYdata1[3]=32;
    m_dbYdata1[4]=32.5;
    m_dbYdata1[5]=33;
    m_dbYdata1[6]=33.6;
    m_dbYdata1[7]=34.7;
    m_dbYdata1[8]=34.9;
    m_dbYdata1[9]=36;
    m_dbYdata2[0]=22.5;
    m_dbYdata2[1]=23.5;
    m_dbYdata2[2]=33.1;
    m_dbYdata2[3]=32.2;
    m_dbYdata2[4]=32.5;
    m_dbYdata2[5]=33.9;
    m_dbYdata2[6]=34.5;
    m_dbYdata2[7]=35.8;
    m_dbYdata2[8]=37.2;
    m_dbYdata2[9]=39;
    return TRUE;
}
```

运行结果：


Untitled - qqq
文件(F) 编辑(E) 查看(V) 帮助(H)
Dispalcement
38.15
35.66
33.16
30.67
28.17
25.68
23.18
20.68
0.12
1.33
2.54
3.75
4.96
6.17
7.38
8.59
Force
数字

第十三章 对话框

1. 签名留念簿

题目要求：签名留念簿程序。该程序模仿签名簿，用户单击窗口客户区后会弹出一个对话框，输入姓名后可在鼠标单击位置显示出该签名。签名的颜色，字体大小和方向随机确定。

程序说明：建立 MFC 单文档，添加一个对话框模板资源。修改对话框模板的 ID 为 **IDD_NAMEDLG**，名字为“**签名对话框**”，并添加一个静态文本控件（签名）和一个编辑控件（**ID_EDITNAME**）。再创建鼠标左键消息函数，并调用 **DoModal()** 函数，完成对话框的弹出。

模块组成：编辑控件，视图客户区实现，随机字体，鼠标左键消息响应

程序分模块实现：

```
//对话框类头文件修改
class CNameDlg : public CDialog
{
// Construction
public:
    CNameDlg(CWnd* pParent = NULL);   // standard constructor
    CPoint m_pointTopLeft;
// Dialog Data
    //{{AFX_DATA(CNameDlg)
    enum { IDD = IDD_NAMEDLG };
    CString m_strNameEdit;
// NOTE: the ClassWizard will add data members here
//}}AFX_DATA


// Overrides
    // ClassWizard generated virtual function overrides
    //{{AFX_VIRTUAL(CNameDlg)
    protected:
    virtual void DoDataExchange(CDataExchange* pDX);    // DDX/DDV support
    //}}AFX_VIRTUAL
```

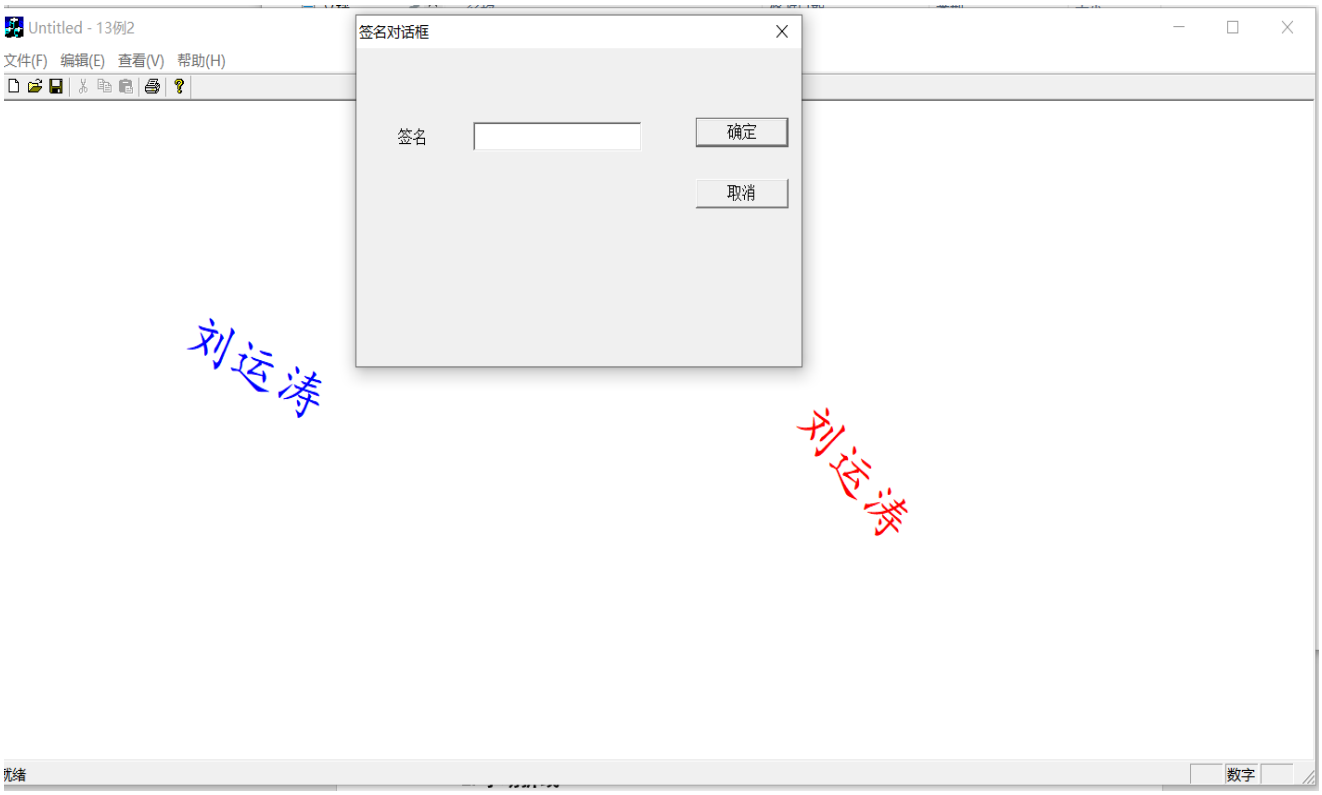
```
// Implementation
protected:

    // Generated message map functions
    //{{AFX_MSG(CNameDlg)
    virtual BOOL OnInitDialog();
    //}}AFX_MSG
    DECLARE_MESSAGE_MAP()
};
//对话框类源文件修改
//对话框类构造函数
CNameDlg::CNameDlg(CWnd* pParent /*=NULL*/)
    : CDialog(CNameDlg::IDD, pParent)
{
//{{AFX_DATA_INIT(CNameDlg)
    m_strNameEdit= T("");
// NOTE: the ClassWizard will add member initialization here
//}}AFX_DATA_INIT
}
//数据交换和数据检验，系统自动生成
void CNameDlg::DoDataExchange(CDataExchange* pDX)
{
    CDialog::DoDataExchange(pDX);
    //{{AFX_DATA_MAP(CNameDlg)
    DDX_Text(pDX,IDC_EDITNAME,m_strNameEdit);
    DDV_MaxChars(pDX,m_strNameEdit,20);
    // NOTE: the ClassWizard will add DDX and DDV calls here
    //}}AFX_DATA_MAP
}
//初始化对话框
BOOL CNameDlg::OnInitDialog()
{
    CDialog::OnInitDialog();

    // TODO: Add extra initialization here
    CRect rect;
    GetWindowRect(&rect);
    rect=CRect(m_pointTopLeft,rect.Size());
    MoveWindow(rect);
    return TRUE;  // return TRUE unless you set the focus to a control
                  // EXCEPTION: OCX Property Pages should return FALSE
}
//签名类
class CSignal : public CObject
```

```
{private:
    CString m_sName;//姓名
    CPoint m_pointSignal;//签名位置
    int m_nHeight;//字体高
    int m_nColor;//签名颜色
    int m_nEscapement;//签名斜角
public:
    CSignal();
    void ShowSignal(CDC*pDC);
    void SetValue(CString name,CPoint point,int height,int color,int escapement);
    virtual ~CSignal();
};
//签名类成员函数
void CSignal::SetValue(CString name,CPoint point,int height,int color,int escapement)
{  m_sName=name;
   m_pointSignal=point;
   m_nHeight=height;
   m_nColor=color;
   m_nEscapement=escapement;
}
//显示签名
void CSignal::ShowSignal(CDC*pDC)
{  CFont*pOldFont,font;
   font.CreateFont(m_nHeight,0,m_nEscapement,0,400,FALSE,FALSE,
                   0,0,OUT_DEFAULT_PRECIS,
                     CLIP_DEFAULT_PRECIS,DEFAULT_QUALITY,
                     DEFAULT_PITCH,"楷体");//创建字体对象
   pOldFont=pDC->SelectObject(&font);
   switch(m_nColor)
   {    case 0:pDC->SetTextColor(RGB(0,0,0));break;
        case 1:pDC->SetTextColor(RGB(255,0,0));break;
        case 2:pDC->SetTextColor(RGB(0,255,0));break;
        case 3:pDC->SetTextColor(RGB(0,0,255));break;
   }
   pDC->TextOut(m_pointSignal.x,m_pointSignal.y,m_sName);
   pDC->SelectObject(pOldFont);
}
//文档类
//在文档类头文件的开始处将签名类的声明包括进来
#if _MSC_VER > 1000
#pragma once
#endif // _MSC_VER > 1000
#include "Signal.h"
#define MAX_NAME 250 //定义签名类对象数组的最大个数
```

```
class CMy132Doc : public CDocument
{
protected: // create from serialization only
    CMy132Doc();
    DECLARE_DYNCREATE(CMy132Doc)

// Attributes
public:
    CSignal m_signalList[MAX_NAME];
    int m_nCount;
// Operations
public:

// Overrides
    // ClassWizard generated virtual function overrides
    //{{AFX_VIRTUAL(CMy132Doc)
    public:
    virtual BOOL OnNewDocument();
    virtual void Serialize(CArchive& ar);
    //}}AFX_VIRTUAL

// Implementation
public:
    virtual ~CMy132Doc();
#ifdef _DEBUG
    virtual void AssertValid() const;
    virtual void Dump(CDumpContext& dc) const;
#endif

protected:

// Generated message map functions
protected:
    //{{AFX_MSG(CMy132Doc)
        // NOTE - the ClassWizard will add and remove member functions here.
        //    DO NOT EDIT what you see in these blocks of generated code !
    //}}AFX_MSG
    DECLARE_MESSAGE_MAP()
};
//文档类 OnNewDocument 虚函数
BOOL CMy132Doc::OnNewDocument()
{
    if (!CDocument::OnNewDocument())
        return FALSE;
```

```
    // TODO: add reinitialization code here
    // (SDI documents will reuse this document)
    m_nCount=0;
    return TRUE;
}
//在视图类的实现文件添加对话框类的头文件并建立鼠标左键消息映射
#include "stdafx.h"
#include "13例2.h"
#include "13例2Doc.h"
#include "13例2View.h"
#include "Signal.h"
#include "NameDlg.h"
//鼠标左键消息响应函数
void CMy132View::OnLButtonDown(UINT nFlags, CPoint point)
{   CMy132Doc*pDoc=GetDocument();
    ASSERT_VALID(pDoc);
    // TODO: Add your message handler code here and/or call default
    if(pDoc->m_nCount<MAX_NAME)
    {   CNameDlg dlg;
        dlg.m_pointTopLeft=point;
        if(dlg.DoModal()==IDOK)
        {   int height=rand()%60+12;
            int color=rand()%4;
            int escapement=(rand()%1200)-600;
            CString name=dlg.m_strNameEdit;

    pDoc->m_signalList[pDoc->m_nCount].SetValue(name,point,height,color,escapement);
            pDoc->m_nCount++;
            Invalidate();
        }
    }
    CView::OnLButtonDown(nFlags, point);
}
//绘制视图客户区函数
void CMy132View::OnDraw(CDC* pDC)
{
    CMy132Doc* pDoc = GetDocument();
    ASSERT_VALID(pDoc);
    // TODO: add draw code for native data here
    for(int i=0;i<pDoc->m_nCount;i++)
       pDoc->m_signalList[i].ShowSignal(pDC);
}
```

运行结果：



2. 手动折线

题目要求：通过输入数据画出折线图。

程序说明：利用菜单编辑器添加一个“输入数据”菜单项，并撤销 pop-up 属性选项。再利用资源编辑器创建一个新的对话框，打开类向导，为新建对话框取名 **MyDlg**，为每一个文本编辑框添加一个成员变量，变量类型为 **double**，变量名依此为 **m_dbX1-m_dbX10, m_dbY1-m_dbY10**，在视图类中添加“输入数据”消息处理函数。

模块组成：输入数据，折线显示，轴标题及刻度，坐标轴

程序分模块实现：

```
//在 MyDoc 类的定义中添加成员变量
class CAsawDoc : public CDocument
{
protected: // create from serialization only
    CAsawDoc();
    DECLARE_DYNCREATE(CAsawDoc)

// Attributes
public:

// Operations
public:

// Overrides
    // ClassWizard generated virtual function overrides
    //{{AFX_VIRTUAL(CAsawDoc)
    public:
    virtual BOOL OnNewDocument();
    virtual void Serialize(CArchive& ar);
    //}}AFX_VIRTUAL

// Implementation
public:
    enum {iPt=10};
    double m_dbXdata[iPt];
    double m_dbYdata[iPt];
```

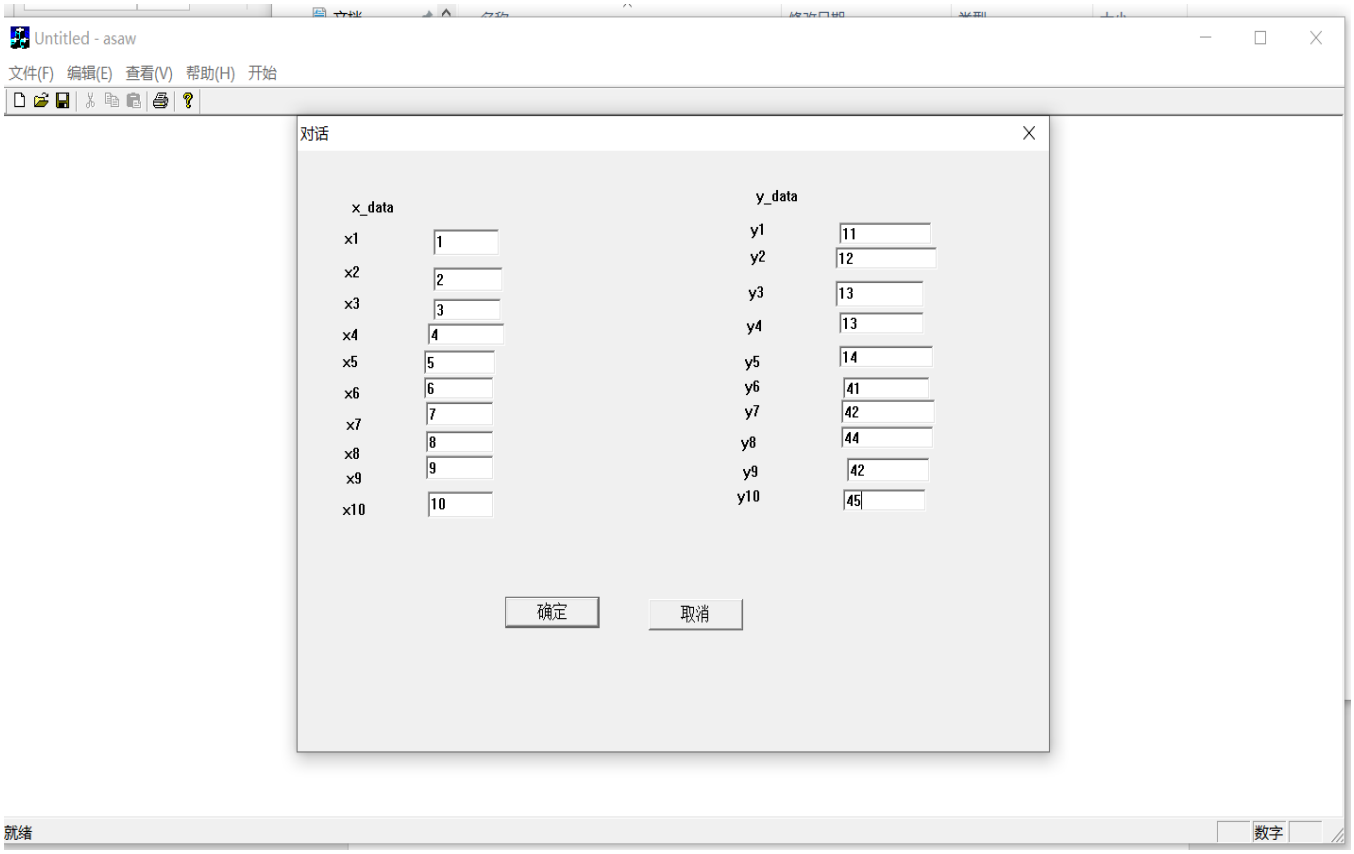
```
    BOOL m_boolNew;
    virtual ~CAsawDoc();
#ifdef _DEBUG
    virtual void AssertValid() const;
    virtual void Dump(CDumpContext& dc) const;
#endif

protected:

// Generated message map functions
protected:
    //{{AFX_MSG(CAsawDoc)
        // NOTE - the ClassWizard will add and remove member functions here.
        //    DO NOT EDIT what you see in these blocks of generated code !
    //}}AFX_MSG
    DECLARE_MESSAGE_MAP()
};
//在 CAsawDoc 的成员函数 OnNewDocument()中初始化数据
BOOL CAsawDoc::OnNewDocument()
{
    if (!CDocument::OnNewDocument())
        return FALSE;

    // TODO: add reinitialization code here
    // (SDI documents will reuse this document)
    for(int i=0;i<iPt;++i)
    {
        m_dbXdata[i]=0;
        m_dbYdata[i]=0;
    }
    m_boolNew=true;
    return TRUE;
}
//在视图文件的开始处将新的对话框的声明包括进来
#include "stdafx.h"
#include "asaw.h"

#include "asawDoc.h"
#include "asawView.h"
#include "MyDlg.h"
#ifdef _DEBUG
#define new DEBUG_NEW
//为“输入数据”菜单的消息处理函数添加代码
void CAsawView::OnInput()
```

```
{CAsawDoc* pDoc = GetDocument();
    ASSERT_VALID(pDoc);
    // TODO: Add your command handler code here
    CMyDlg dlgInput;
    int nRet=dlgInput.DoModal();
    if(nRet==IDOK)
    {
    pDoc->m_dbXdata[0]=dlgInput.m_dbX1;
    pDoc->m_dbXdata[1]=dlgInput.m_dbX2;
    pDoc->m_dbXdata[2]=dlgInput.m_dbX3;
    pDoc->m_dbXdata[3]=dlgInput.m_dbX4;
    pDoc->m_dbXdata[4]=dlgInput.m_dbX5;
    pDoc->m_dbXdata[5]=dlgInput.m_dbX6;
    pDoc->m_dbXdata[6]=dlgInput.m_dbX7;
    pDoc->m_dbXdata[7]=dlgInput.m_dbX8;
    pDoc->m_dbXdata[8]=dlgInput.m_dbX9;
    pDoc->m_dbXdata[9]=dlgInput.m_dbX10;

    pDoc->m_dbYdata[0]=dlgInput.m_dbY1;
    pDoc->m_dbYdata[1]=dlgInput.m_dbY2;
    pDoc->m_dbYdata[2]=dlgInput.m_dbY3;
    pDoc->m_dbYdata[3]=dlgInput.m_dbY4;
    pDoc->m_dbYdata[4]=dlgInput.m_dbY5;
    pDoc->m_dbYdata[5]=dlgInput.m_dbY6;
    pDoc->m_dbYdata[6]=dlgInput.m_dbY7;
    pDoc->m_dbYdata[7]=dlgInput.m_dbY8;
    pDoc->m_dbYdata[8]=dlgInput.m_dbY9;
    pDoc->m_dbYdata[9]=dlgInput.m_dbY10;

    pDoc->m_boolNew=false;
    Invalidate();
  }
}
//编辑 OnDraw 函数显示折线
void CAsawView::OnDraw(CDC* pDC)
{
    CAsawDoc* pDoc = GetDocument();
    ASSERT_VALID(pDoc);
    // TODO: add draw code for native data here
    const int xOrg=100;
    const int yOrg=350;
    const int xMax=700;
    const int yMin=20;
    const int xMin=xOrg;
```

```
const int yMax=yOrg;
if(!pDoc->m_boolNew)
{
 double dbXMin=pDoc->m_dbXdata[0];
 double dbXMax=pDoc->m_dbXdata[0];
 double dbYMin=pDoc->m_dbYdata[0];
 double dbYMax=pDoc->m_dbYdata[0];
for(int i=1;i<pDoc->iPt;i++)
{
     pDoc->m_dbXdata[i]<dbXMin?dbXMin=pDoc->m_dbXdata[i]:
    pDoc->m_dbXdata[i]<dbXMax?dbXMax=pDoc->m_dbXdata[i]:dbXMax;
     pDoc->m_dbYdata[i]<dbYMin?dbYMin=pDoc->m_dbYdata[i]:
    pDoc->m_dbYdata[i]<dbYMax?dbYMax=pDoc->m_dbYdata[i]:dbYMax;
}
dbXMax=dbXMax+(dbXMax-dbXMin)/pDoc->iPt;
dbXMin=dbXMin-(dbXMax-dbXMin)/pDoc->iPt;
dbYMax=dbYMax+(dbYMax-dbYMin)/pDoc->iPt;
dbYMin=dbYMin-(dbYMax-dbYMin)/pDoc->iPt;

double dbXRatio=(dbXMax-dbXMin)<10e-15?10e-15:(xMax-xOrg)/(dbXMax-dbXMin);
double dbYRatio=(dbYMax-dbYMin)<10e-15?10e-15:(yOrg-yMin)/(dbYMax-dbYMin);

int x=(int)(dbXRatio*(pDoc->m_dbXdata[0]-dbXMin)+xOrg);
int y=(int)(yOrg-dbYRatio*(pDoc->m_dbYdata[0]-dbYMin));
     pDC->MoveTo(x,y);
     CBrush brushNewBrush(RGB(0,0,0));
     CBrush * pbrushOldBrush=pDC->SelectObject(&brushNewBrush);
     CRect rectSymbol(x-3,y-3,x+3,y+3);
     pDC->Ellipse(rectSymbol);
     for( i=1;i<pDoc->iPt;i++)
{
  x=(int)(dbXRatio*(pDoc->m_dbXdata[i]-dbXMin)+xOrg);
  y=(int)(yOrg-dbYRatio*(pDoc->m_dbYdata[i]-dbYMin));
   rectSymbol.left=x-3;
   rectSymbol.top=y-3;
   rectSymbol.right=x+3;
   rectSymbol.bottom=y+3;
   pDC->Ellipse(rectSymbol);
   pDC->LineTo(x,y);
}
pDC->SelectObject(pbrushOldBrush);
pDC->MoveTo(xOrg,yOrg);
pDC->LineTo(xMax,yOrg);
pDC->MoveTo(xOrg,yOrg);
```

```
        pDC->LineTo(xOrg,yMin);

    const iNumScal=8;
        double dbXInc=(dbXMax-dbXMin)/iNumScal;
        y=yOrg;
        for(i=0;i<iNumScal;++i)
        {
            x=(int)(dbXRatio*i*dbXInc+xOrg);
            pDC->MoveTo(x,y);
            pDC->LineTo(x,y+8);
            CString strBufOut;
            strBufOut.Format("%6.2f",(dbXMin+i*dbXInc));
            pDC->TextOut(x-20,y+15,strBufOut);
        }
        double dbYInc=(dbYMax-dbYMin)/iNumScal;
        x=xOrg;
        for(i=0;i<iNumScal;++i)
        {
            y=(int)(yOrg-dbYRatio*i*dbYInc);
            pDC->MoveTo(x,y);
            pDC->LineTo(x-8,y);
            CString strBufOut;
            strBufOut.Format("%6.2f",(dbYMin+i*dbYInc));
            pDC->TextOut(x-55,y-8,strBufOut);
        }

        x=(xMax-xOrg)/8*7;
        y=yOrg+40;
        pDC->TextOut(x,y,"X");
        x=xOrg-80;
        y=(yOrg-yMin)/8;
        pDC->TextOut(x,y,"Y");
        }
        else
        {
        CRect rectClient;
        GetClientRect(rectClient);
        pDC->FillSolidRect(rectClient,RGB(255,255,255));
        }
    }
```

运行结果：


Untitled - asaw
文件(F) 编辑(E) 查看(V) 帮助(H) 开始
对话
x_data
x1 1
x2 2
x3 3
x4 4
x5 5
x6 6
x7 7
x8 8
x9 9
x10 10
y_data
y1 11
y2 12
y3 13
y4 13
y5 14
y6 41
y7 42
y8 44
y9 42
y10 45
确定
取消
就绪
数字

3. 彩色泡泡

题目要求：在吹泡泡程序的基础上添加颜色选择对话框，使其可以绘出五颜六色的泡泡。

程序说明：在文档类声明中添加一个 **COLORREF 数组**，存放泡泡颜色，并在添加泡泡前把泡泡的颜色初始化为浅灰色，其他要点与吹泡泡程序基本类似。.

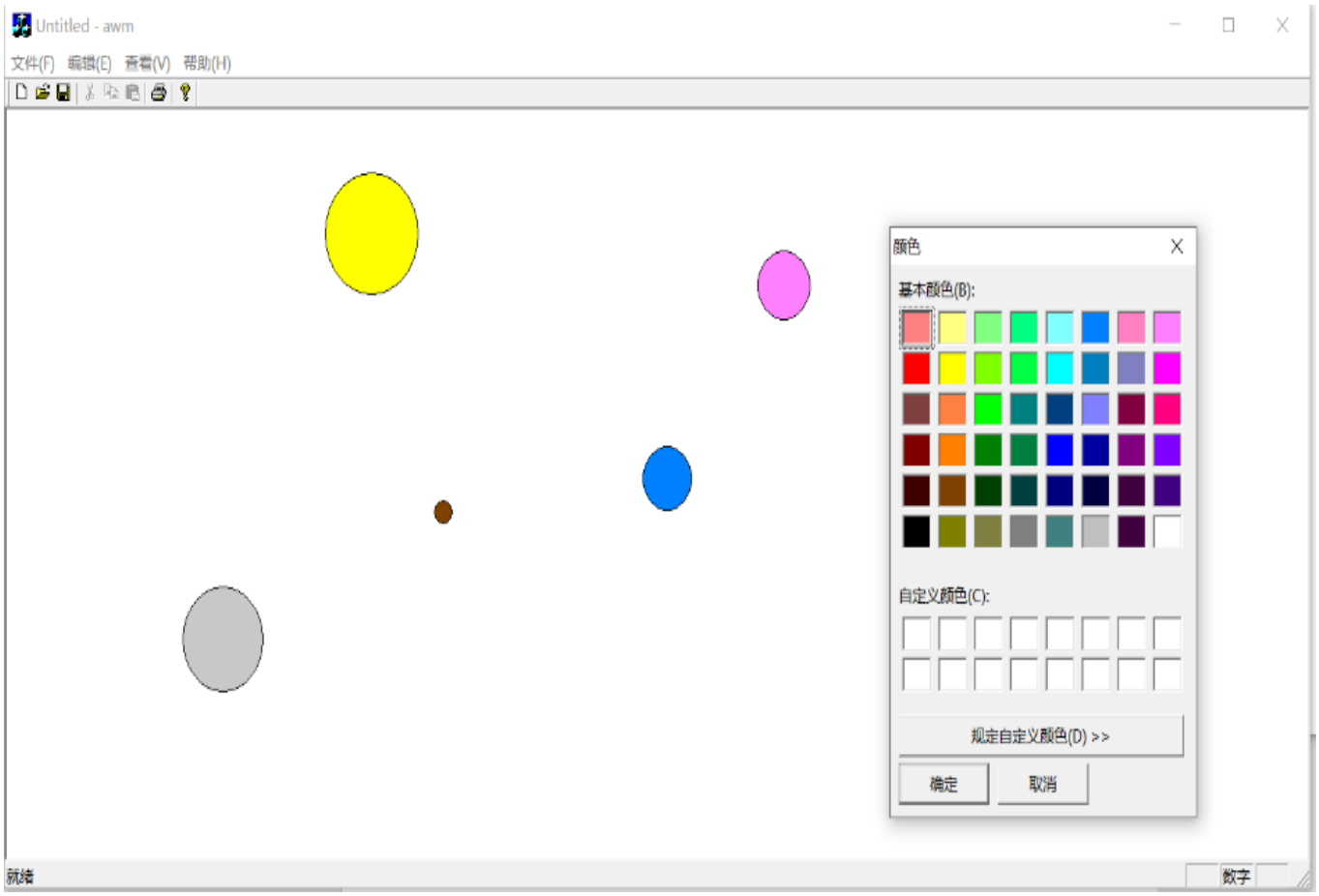
模块组成：颜色数组定义及初始化，泡泡显示，鼠标左键消息响应

程序分模块实现：

```
//在 CAwmDoc 类声明中另添加 COLORREF 数组，存放各泡泡颜色
class CAwmDoc : public CDocument
{
protected: // create from serialization only
    CAwmDoc();
    DECLARE_DYNCREATE(CAwmDoc)

// Attributes
public:
 CRect m_rectBubble[MAX_BUBBLE];
  int m_nBubbleCount;
  COLORREF m_colorBubble[MAX_BUBBLE];
// Operations
public:

// Overrides
    // ClassWizard generated virtual function overrides
    //{{AFX_VIRTUAL(CAwmDoc)
    public:
    virtual BOOL OnNewDocument();
    virtual void Serialize(CArchive& ar);
    //}}AFX_VIRTUAL

// Implementation
public:
    virtual ~CAwmDoc();
#ifdef _DEBUG
    virtual void AssertValid() const;
    virtual void Dump(CDumpContext& dc) const;
#endif
```

```
protected:
// Generated message map functions
protected:
    //{{AFX_MSG(CAwmDoc)
        // NOTE - the ClassWizard will add and remove member functions here.
        //    DO NOT EDIT what you see in these blocks of generated code !
    //}}AFX_MSG
    DECLARE_MESSAGE_MAP()
};
//修改鼠标左键消息映射函数，添加使用颜色选择公共对话框的代码
void CAwmView::OnLButtonDown(UINT nFlags, CPoint point)
{
    // TODO: Add your message handler code here and/or call default
    CAwmDoc* pDoc = GetDocument();
    ASSERT_VALID(pDoc);
    if(pDoc->m_nBubbleCount<MAX_BUBBLE)
    {   pDoc->m_colorBubble[pDoc->m_nBubbleCount]=RGB(200,200,200);
    CColorDialog dlg(pDoc->m_colorBubble[pDoc->m_nBubbleCount]);
    if(dlg.DoModal()==IDOK)
        pDoc->m_colorBubble[pDoc->m_nBubbleCount]=dlg.GetColor();
    int r=rand()%50+10;
    CRect rect(point.x-r,point.y-r,point.x+r,point.y+r);
    pDoc->m_rectBubble[pDoc->m_nBubbleCount]=rect;
    pDoc->m_nBubbleCount++;
    InvalidateRect(rect,FALSE);
    }
    CView::OnLButtonDown(nFlags, point);
}
//修改 OnDraw()成员函数，添加根据泡泡颜色使用的画刷代码，并在显示泡泡后销毁
void CAwmView::OnDraw(CDC* pDC)
{
    CAwmDoc* pDoc = GetDocument();
    ASSERT_VALID(pDoc);
    // TODO: add draw code for native data here
    CBrush brushNew,*pbrushOld;
    for(int i=0;i<pDoc->m_nBubbleCount;i++)
    {
        brushNew.CreateSolidBrush(pDoc->m_colorBubble[i]);
        pbrushOld=pDC->SelectObject(&brushNew);
        pDC->Ellipse(pDoc->m_rectBubble[i]);
        pDC->SelectObject(pbrushOld);
        brushNew.DeleteObject();
    }
}
```

运行结果：


Untitled - awm
文件(F) 编辑(E) 查看(V) 帮助(H)
颜色
基本颜色(B):
自定义颜色(C):
规定自定义颜色(D) >>
确定
取消
就绪
数字

第十四章 控件

1. 人事管理系统

题目要求：为某公司设计一个人事管理系统，其基本功能为输入，编辑和查看公司的人事档案。职工人事档案包括姓名性别，出生日期，婚姻状况，所在部门，职务和工资。

程序说明：首先，生成一个名为 EMP 的 SDI 程序结构，调出高级设置对话框，在文件格式字符串选项卡中将文件后缀名设置为**“dat”**，在窗口设置选项卡中设置最大化检查框为**“选中”**状态。然后，创建一个对话框模板，将其标识符改为**IDD_EMPLOYEE**,并适当调整其大小，再向里面加入以下控件：

静态文本控件：姓名，出生日期，部门，职务，工资。

三个编辑控件，其 ID 分别改为 **IDC_NAME，IDC_BIRTHDAY 和 IDC_SALARY** 。

两个单选按钮，其 ID 和名字分 **IDC_MALE，“男“和 IDC_FEMALE，”女“**。并为 IDC_MALE 设置属性 **Group**。

一个检查框，其 ID 改为 **IDC_POSITION**，保证 Styles 选项卡中的 Selection 项为 **Single**，并将 Sort 检查框设置为非选中状态。

一个组合框，其 ID 改为 **IDC_POSITION**，并在 Styles 选项卡中将 Type 设置为 **Drop List**,并将 Sort 检查框设置为非选中状态。然后在 Data 选项卡中输入各种职务名称（总经理，副总经理，部门经理，项目经理，业务经理）。

一个组框，将其名字改为“**个人资料**“，调节其大小和位置，使之可以框住姓名，性别，出生日期和婚姻状况等控件。

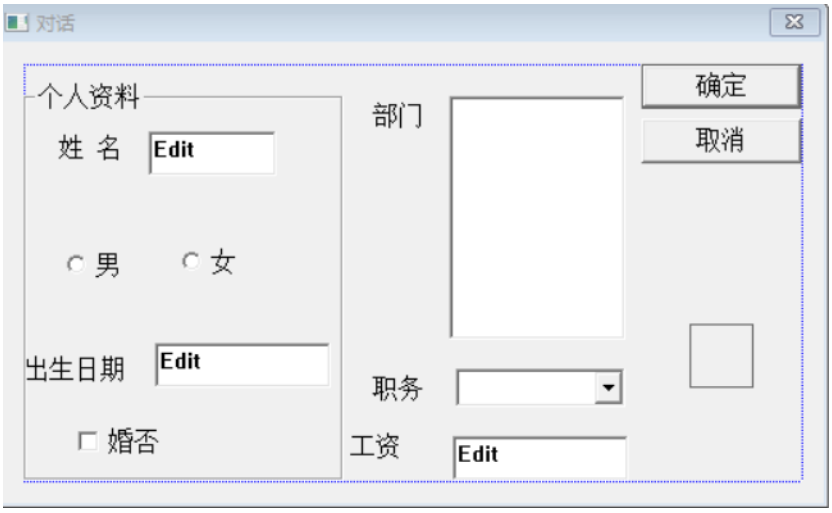
再之后为各控件设置 **Tab Order**，并创建对话框类，类名为 **CEmpDlg**。利用类向导，为对话框类添加与各控件对应的数据成员，其名字，类型和 ID 分别为：

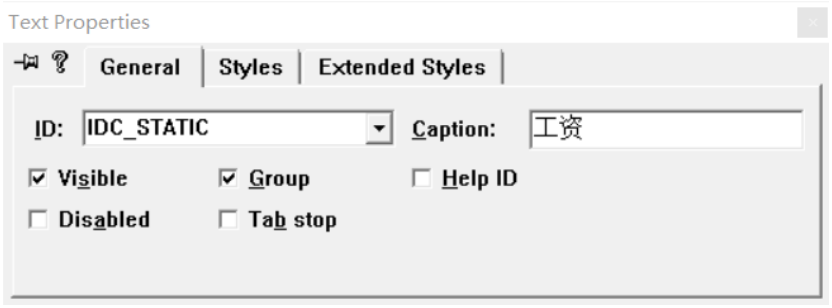
Control IDs	Variable Type	Member variable name
IDC_BIRTHDAY	COleDateTime	m_tBirthday
IDC_DEPT	CString	m_atrDept
IDC_MALE	int	m_nSex
IDC_MARRIED	BOOL	m_bMarried
IDC_NAME	CString	m_strName
IDC_POSITION	CString	m_strPosition
IDC_SALARY	float	m_fSalary

接着，还应编辑项目的菜单资源，在框架窗口的主菜单（IDC_MAINFRAME）中添加两个菜单选项，并未下拉菜单“编辑“添加 3 个菜单选项:

ID	Caption	prompt
ID_NEXT	下一记录	下一职工档案\n 下一记录
ID_PREV	上一记录	上一职工档案\n 上一记录
ID_APPEND	输入（&I）/tCtrl+I	输入新职工档案\n 输入
ID_EDIT	输入（&E）/tCtrl+E	编辑职工档案\n 编辑
ID_DELETE	输入（&D）/tCtrl+D	删除职工档案\n 删除

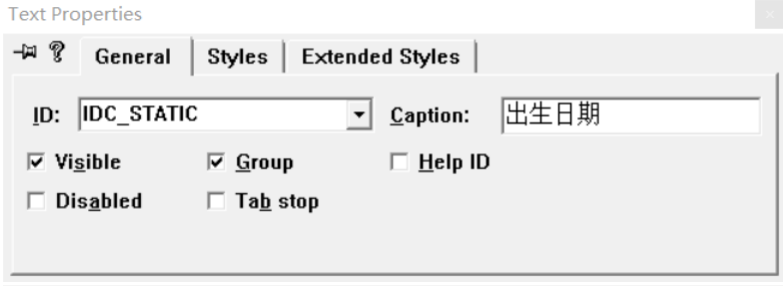
然后再次利用类向导为视图类添加与这些菜单选项对应的成员函数。最后为项目添加职工类 **CEmployee**,其父类为 **Cobject**。下面附具体操作流程:






Text Properties
General
Styles
Extended Styles
ID: IDC_STATIC
Caption: 职务
Visible
Group
Help ID
Disabled
Tab stop

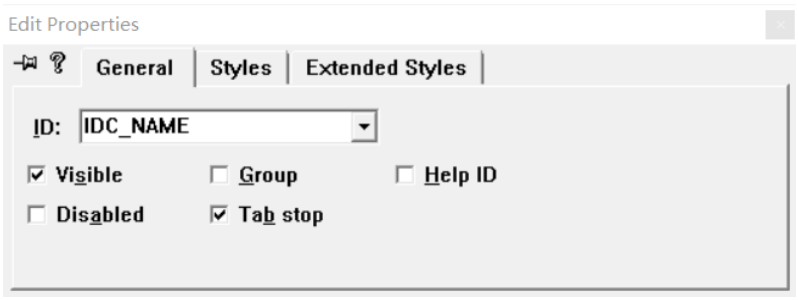

Text Properties
General
Styles
Extended Styles
ID: IDC_STATIC
Caption: 部门
Visible
Group
Help ID
Disabled
Tab stop

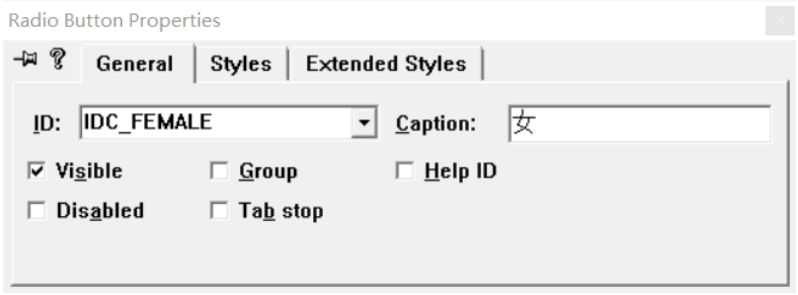

Text Properties
General
Styles
Extended Styles
ID: IDC_STATIC
Caption: 出生日期
Visible
Group
Help ID
Disabled
Tab stop

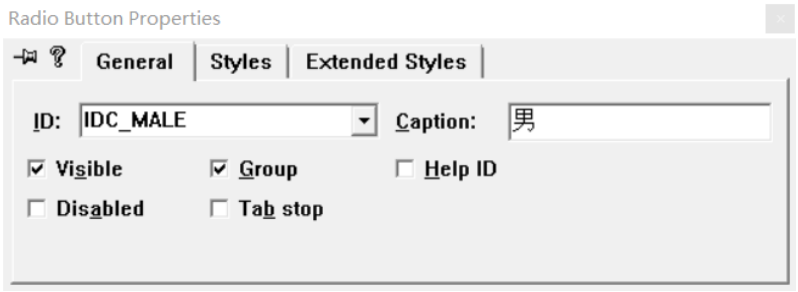

Text Properties
General
Styles
Extended Styles
ID: IDC_STATIC
Caption: 姓 名
Visible
Group
Help ID
Disabled
Tab stop


Edit Properties
General
Styles
Extended Styles
ID: IDC_SALARY
Visible
Group
Help ID
Disabled
Tab stop

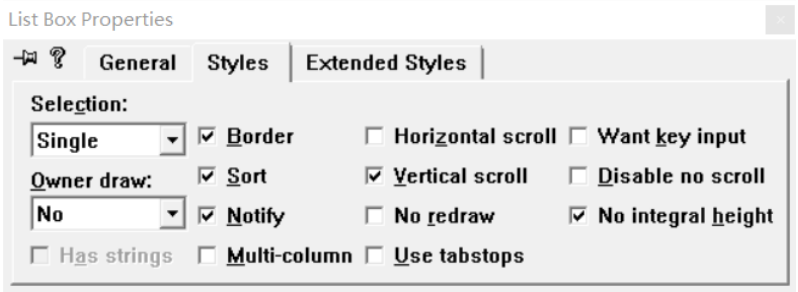

Edit Properties
General
Styles
Extended Styles
ID: IDC_BIRTHDATE
Visible
Group
Help ID
Disabled
Tab stop

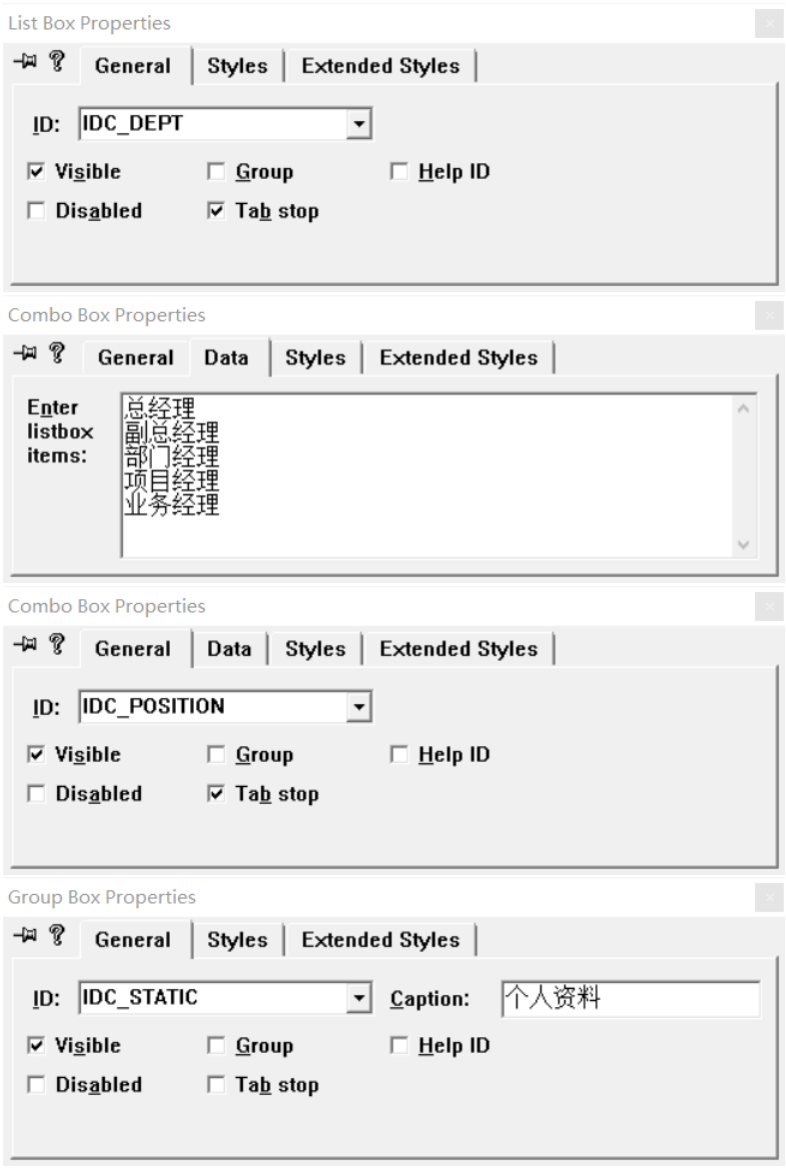

Edit Properties
General
Styles
Extended Styles
ID: IDC_NAME
Visible
Group
Help ID
Disabled
Tab stop


Radio Button Properties
General
Styles
Extended Styles
ID: IDC_FEMALE
Caption: 女
Visible
Group
Help ID
Disabled
Tab stop


Radio Button Properties
General
Styles
Extended Styles
ID: IDC_MALE
Caption: 男
Visible
Group
Help ID
Disabled
Tab stop


Check Box Properties
General
Styles
Extended Styles
ID: IDC_MARRIED
Caption: 婚否
Visible
Group
Help ID
Disabled
Tab stop


List Box Properties
General
Styles
Extended Styles
Selection:
Single
Owner draw:
No
Border
Sort
Notify
Horizontal scroll
Vertical scroll
No redraw
Want key input
Disable no scroll
No integral height
Has strings
Multi-column
Use tabstops



模块组成：基本对话框，控件，菜单，菜单选项

程序分模块实现：

//重载 CEmpDlg 类的 OnInitDialog()成员函数

```
BOOL CEmoDlg::OnInitDialog()
{
    CDialog::OnInitDialog();

    // TODO: Add extra initialization here
    CListBox *pLB=(CListBox *)GetDlgItem(IDC_DEPT);
    pLB->InsertString(-1,"办公室");
    pLB->InsertString(-1,"开发部");
    pLB->InsertString(-1,"生产部");
    pLB->InsertString(-1,"销售部");
    pLB->InsertString(-1,"人事部");
```

```
    return CDialog::OnInitDialog();
    return TRUE;  // return TRUE unless you set the focus to a control
                  // EXCEPTION: OCX Property Pages should return FALSE
}
```

//CEmployee 类定义

```
class CEmoDlg : public CDialog
{
// Construction
public:
    CEmoDlg(CWnd* pParent = NULL);   // standard constructor

// Dialog Data
    //{{AFX_DATA(CEmoDlg)
    enum { IDD = IDD_EMPLOYEE };
    COleDateTime m_tBirthday;
    CString  m_strDept;
    float    m_fSalary;
    CString  m_strPosition;
    CString  m_strName;
    BOOL m_bMarried;
    int     m_nSex;
    //}}AFX_DATA


// Overrides
    // ClassWizard generated virtual function overrides
    //{{AFX_VIRTUAL(CEmoDlg)
    protected:
    virtual void DoDataExchange(CDataExchange* pDX);    // DDX/DDV support
    //}}AFX_VIRTUAL

// Implementation
protected:

    // Generated message map functions
    //{{AFX_MSG(CEmoDlg)
    virtual BOOL OnInitDialog();
    //}}AFX_MSG
    DECLARE_MESSAGE_MAP()
};
```

//在文档类的头文件首部添加 CEmployee 类的头文件以及一个 CEmployee 类的数组

```
#if _MSC_VER > 1000
#pragma once
#endif // _MSC_VER > 1000
```

```
#include "Employee1.h"
#define MAX_EMPLOYEE 1000

class CEeeDoc : public CDocument
{
protected: // create from serialization only
    CEeeDoc();
    DECLARE_DYNCREATE(CEeeDoc)

// Attributes
public:
    CEmployee1 m_empList[MAX_EMPLOYEE];
    int m_nCount;
// Operations
public:

// Overrides
    // ClassWizard generated virtual function overrides
    //{{AFX_VIRTUAL(CEeeDoc)
    public:
    virtual BOOL OnNewDocument();
    virtual void Serialize(CArchive& ar);
    virtual void DeleteContents();
    //}}AFX_VIRTUAL

// Implementation
public:
    virtual ~CEeeDoc();
#ifdef _DEBUG
    virtual void AssertValid() const;
    virtual void Dump(CDumpContext& dc) const;
#endif

protected:

// Generated message map functions
protected:
    //{{AFX_MSG(CEeeDoc)
        // NOTE - the ClassWizard will add and remove member functions here.
        //    DO NOT EDIT what you see in these blocks of generated code !
    //}}AFX_MSG
    DECLARE_MESSAGE_MAP()
};
```

```
//重载成员函数 DeleteContents()
void CEeeDoc::DeleteContents()
{
    // TODO: Add your specialized code here and/or call the base class
    m_nCount=0;
    CDocument::DeleteContents();
}
//在视图类定义中添加一个用于记录当前操作记录的数据成员
class CEeeView : public CView
{
protected: // create from serialization only
    CEeeView();
    DECLARE_DYNCREATE(CEeeView)

// Attributes
public:
    CEeeDoc* GetDocument();

// Operations
public:

// Overrides
    // ClassWizard generated virtual function overrides
    //{{AFX_VIRTUAL(CEeeView)
    public:
    virtual void OnDraw(CDC* pDC);  // overridden to draw this view
    virtual BOOL PreCreateWindow(CREATESTRUCT& cs);
    virtual void OnInitialUpdate();
    protected:
    virtual BOOL OnPreparePrinting(CPrintInfo* pInfo);
    virtual void OnBeginPrinting(CDC* pDC, CPrintInfo* pInfo);
    virtual void OnEndPrinting(CDC* pDC, CPrintInfo* pInfo);
    //}}AFX_VIRTUAL

// Implementation
public:
    int m_nCurrEmp;
    virtual ~CEeeView();
#ifdef _DEBUG
    virtual void AssertValid() const;
    virtual void Dump(CDumpContext& dc) const;
#endif

protected:
```

```
// Generated message map functions
protected:
    //{{AFX_MSG(CEeeView)
    afx_msg void OnDelete();
    afx_msg void OnEdit();
    afx_msg void OnPrev();
    afx_msg void OnNext();
    afx_msg void OnAppend();
    //}}AFX_MSG
    DECLARE_MESSAGE_MAP()
};
//重载视图类 OnInitialUpdate()函数
void CEeeView::OnInitialUpdate()
{
    CView::OnInitialUpdate();

    // TODO: Add your specialized code here and/or call the base class
    CEeeView::OnInitialUpdate();
    m_nCurrEmp=0;
    Invalidate();
}
//视图类的 OnDraw()成员函数用于显示正在操作的职工档案
void CEeeView::OnDraw(CDC* pDC)
{
    CEeeDoc* pDoc = GetDocument();
    ASSERT_VALID(pDoc);
    // TODO: add draw code for native data here
    CString s;
    s.Format("职工人数:%d",pDoc->m_nCount);
    pDC->SetTextColor(RGB(255,0,0));
    pDC->TextOut(40,40,s);
    s.Format("职工编号:%d",m_nCurrEmp+1);
    pDC->TextOut(340,40,s);
    pDC->MoveTo(40,70);
    pDC->LineTo(600,70);
    if (pDoc->m_nCount>0)
    {
        pDC->SetTextColor(RGB(0,0,0));
        pDC->TextOut(140, 90,"姓名");
        pDC->TextOut(140,  130,"性别");
        pDC->TextOut(140,  170,"出生日期");
        pDC->TextOut(140,  210,"婚姻状态");
        pDC->TextOut(140,  250,"部门");
```

```
        pDC->TextOut(140,   290,"职务");
        pDC->TextOut(140,   330,"工资");


        pDC->SetTextColor(RGB(0,0,255));
        pDC->TextOut(300,90,pDoc->m_empList[m_nCurrEmp].m_strName);

        if(pDoc->m_empList[m_nCurrEmp].m_nSex==0)
            pDC->TextOut(300,130,"男");
        else
            pDC->TextOut(300,130,"女");
        s=pDoc->m_empList[m_nCurrEmp].m_tBirthday.Format("%Y.%m.%d");
        pDC->TextOut(300,170,s);
        if(pDoc->m_empList[m_nCurrEmp].m_bMarried)
            pDC->TextOut(300,210,"已婚");
        else
            pDC->TextOut(300,210,"未婚");
        pDC->TextOut(300,250,pDoc->m_empList[m_nCurrEmp].m_strDept);
        pDC->TextOut(300,290,pDoc->m_empList[m_nCurrEmp].m_strPosition);
        s.Format("%8.2f",pDoc->m_empList[m_nCurrEmp].m_fSalary);
        pDC->TextOut(300,330,s);
    }

}
//菜单消息响应函数
void CEeeView::OnAppend()
{CEeeDoc* pDoc = GetDocument();
    ASSERT_VALID(pDoc);
    // TODO: Add your command handler code here
    CEmoDlg dlg;
    if(dlg.DoModal()==IDOK)
    {
        pDoc->m_nCount++;
           m_nCurrEmp=pDoc->m_nCount-1;
            pDoc->m_empList[m_nCurrEmp].m_strName=dlg.m_strName;
            pDoc->m_empList[m_nCurrEmp].m_nSex=dlg.m_nSex;
            pDoc->m_empList[m_nCurrEmp].m_tBirthday=dlg.m_tBirthday;
            pDoc->m_empList[m_nCurrEmp].m_bMarried=dlg.m_bMarried;
            pDoc->m_empList[m_nCurrEmp].m_strDept=dlg.m_strDept;
            pDoc->m_empList[m_nCurrEmp].m_strPosition=dlg.m_strPosition;
            pDoc->m_empList[m_nCurrEmp].m_fSalary=dlg.m_fSalary;
        pDoc->SetModifiedFlag();
         Invalidate();
    }
```

```
}
void CEeeView::OnDelete()
{CEeeDoc* pDoc = GetDocument();
    ASSERT_VALID(pDoc);
    // TODO: Add your command handler code here
    if (pDoc->m_nCount)
    {
        for (int i=m_nCurrEmp;i<pDoc->m_nCount-1;i++)
        {
            pDoc->m_empList[i].m_bMarried=pDoc->m_empList[i+1].m_bMarried;
            pDoc->m_empList[i].m_fSalary=pDoc->m_empList[i+1].m_fSalary;
            pDoc->m_empList[i].m_nSex=pDoc->m_empList[i+1].m_nSex;
            pDoc->m_empList[i].m_strDept=pDoc->m_empList[i+1].m_strDept;
            pDoc->m_empList[i].m_strName=pDoc->m_empList[i+1].m_strName;
            pDoc->m_empList[i].m_strPosition=pDoc->m_empList[i+1].m_strPosition;
            pDoc->m_empList[i].m_tBirthday=pDoc->m_empList[i+1].m_tBirthday;
        }
        pDoc->m_nCount--;
        if(m_nCurrEmp>pDoc->m_nCount-1)
            m_nCurrEmp=pDoc->m_nCount-1;
        pDoc->SetModifiedFlag();
        Invalidate();
    }
}
void CEeeView::OnEdit()
{CEeeDoc* pDoc = GetDocument();
    ASSERT_VALID(pDoc);
    // TODO: Add your command handler code here
    if (pDoc->m_nCount)
    {
        CEmoDlg dlg;
        dlg.m_strName=pDoc->m_empList[m_nCurrEmp].m_strName;
        dlg.m_nSex=pDoc->m_empList[m_nCurrEmp].m_nSex;
        dlg.m_tBirthday=pDoc->m_empList[m_nCurrEmp].m_tBirthday;
        dlg.m_bMarried=pDoc->m_empList[m_nCurrEmp].m_bMarried;
        dlg.m_strDept=pDoc->m_empList[m_nCurrEmp].m_strDept;
        dlg.m_strPosition=pDoc->m_empList[m_nCurrEmp].m_strPosition;
        dlg.m_fSalary=pDoc->m_empList[m_nCurrEmp].m_fSalary;
        if (dlg.DoModal()==IDOK)
        {
            pDoc->m_empList[m_nCurrEmp].m_strName=dlg.m_strName;
            pDoc->m_empList[m_nCurrEmp].m_nSex=dlg.m_nSex;
            pDoc->m_empList[m_nCurrEmp].m_tBirthday=dlg.m_tBirthday;
```

```
            pDoc->m_empList[m_nCurrEmp].m_bMarried=dlg.m_bMarried;
            pDoc->m_empList[m_nCurrEmp].m_strDept=dlg.m_strDept;
            pDoc->m_empList[m_nCurrEmp].m_strPosition=dlg.m_strPosition;
            pDoc->m_empList[m_nCurrEmp].m_fSalary=dlg.m_fSalary;
            Invalidate();
        }
    }
}
void CEeeView::OnNext()
{
    // TODO: Add your command handler code here
    CEeeDoc* pDoc = GetDocument();
    ASSERT_VALID(pDoc);
if(pDoc->m_nCount>1)
{
  if(m_nCurrEmp==pDoc->m_nCount-1)
  m_nCurrEmp==0;
else
  m_nCurrEmp++;
}
    Invalidate();
}
void CEeeView::OnPrev()
{
    // TODO: Add your command handler code here
    CEeeDoc* pDoc = GetDocument();
    ASSERT_VALID(pDoc);
    if(pDoc->m_nCount>1)
    {if(m_nCurrEmp==0)
    m_nCurrEmp=pDoc->m_nCount-1;
    else
        m_nCurrEmp--;
    }
    Invalidate();
}
```

运行结果：


Untitled - 141
文件(F) 编辑(E) 查看(V) 帮助(H) 菜单(M)Ctrl+M
职工人数：0
职工编号：1
女
1899.12.30
已婚
-431602080.00
对话
个人资料
姓名
男
女
出生日期
2017/9/12 1
婚否
部门
办公室
开发部
生产部
销售部
人事部
职务
工资
0
确定
取消
就绪
数字

2. 自定义彩色泡泡

题目要求：用基于对话框的应用程序结构实现彩色吹泡泡程序。由于对话框本身结构简单，没有明显的客户区，颜色也不醒目，所以在对话框上自行建立一个矩形区域作为吹泡泡的客户区，并通过一个“颜色设置”按钮来设置泡泡的颜色。

程序说明：创建一个基于对话框的应用程序框架，所有设置均使用默认值。使用使用对话框模板编辑器编辑作为主界面窗口的对话框模板,删除其中的静态文本控件和 Cancel 按钮,将 OK 按钮的 Caption 设置为“**完成**”,并将对话框大小调整为 400×300 左右。为对话框模板添加一个 Picture 控件,将其 ID 改为 **IDC_CLIENT**,将其 Type 设置为 **Frame**, Color 设置 **Black**,并设置 Sunken 属性。调整其位置为 **(7,7)**，大小为 **287×287**。这个框中即为自定义的吹泡泡客户区,所有的吹泡泡活动均在该区域中进行。为对话框模板添加一个按钮,将其 ID 改为 **IDC_ COLOR**, Caption 改为“**颜色设置**”。使用类向导为对话框类加一个鼠标左键消息响应函数 **OnLButtonDown()**和一个按钮命令消息响应函数 **OnColor()**。

模块组成：泡泡显示，色彩设置，鼠标左键消息响应，基本对话框

程序分模块实现：

```
//在对话框类定义中添加存放泡泡的几何参数和颜色的数组数据成员
#define MAX_BUBBLE 250
class CYouDlg : public CDialog
{
// Construction
public:
    CYouDlg(CWnd* pParent = NULL); // standard constructor
    CRect m_rectBubble[MAX_BUBBLE];
    COLORREF m_colorBubble[MAX_BUBBLE];
    int m_nBubbleCount;
    CRect m_rectClient;
    COLORREF m_colorCurrent;
// Dialog Data
    //{{AFX_DATA(CYouDlg)
```

```
    enum { IDD = IDD_YOU_DIALOG };
        // NOTE: the ClassWizard will add data members here
    //}}AFX_DATA

    // ClassWizard generated virtual function overrides
    //{{AFX_VIRTUAL(CYouDlg)
    protected:
    virtual void DoDataExchange(CDataExchange* pDX); // DDX/DDV support
    //}}AFX_VIRTUAL

// Implementation
protected:
    HICON m_hIcon;

    // Generated message map functions
    //{{AFX_MSG(CYouDlg)
    virtual BOOL OnInitDialog();
    afx_msg void OnSysCommand(UINT nID, LPARAM lParam);
    afx_msg void OnPaint();
    afx_msg HCURSOR OnQueryDragIcon();
    afx_msg void OnLButtonDown(UINT nFlags, CPoint point);
    afx_msg void OnColor();
    //}}AFX_MSG
    DECLARE_MESSAGE_MAP()
};
//修改对话框类的 OnInitDialog()成员函数
BOOL CYouDlg::OnInitDialog()
{
    CDialog::OnInitDialog();

    // Add "About..." menu item to system menu.

    // IDM_ABOUTBOX must be in the system command range.
    ASSERT((IDM_ABOUTBOX & 0xFFF0) == IDM_ABOUTBOX);
    ASSERT(IDM_ABOUTBOX < 0xF000);

    CMenu* pSysMenu = GetSystemMenu(FALSE);
    if (pSysMenu != NULL)
    {
        CString strAboutMenu;
        strAboutMenu.LoadString(IDS_ABOUTBOX);
        if (!strAboutMenu.IsEmpty())
        {
            pSysMenu->AppendMenu(MF_SEPARATOR);
```

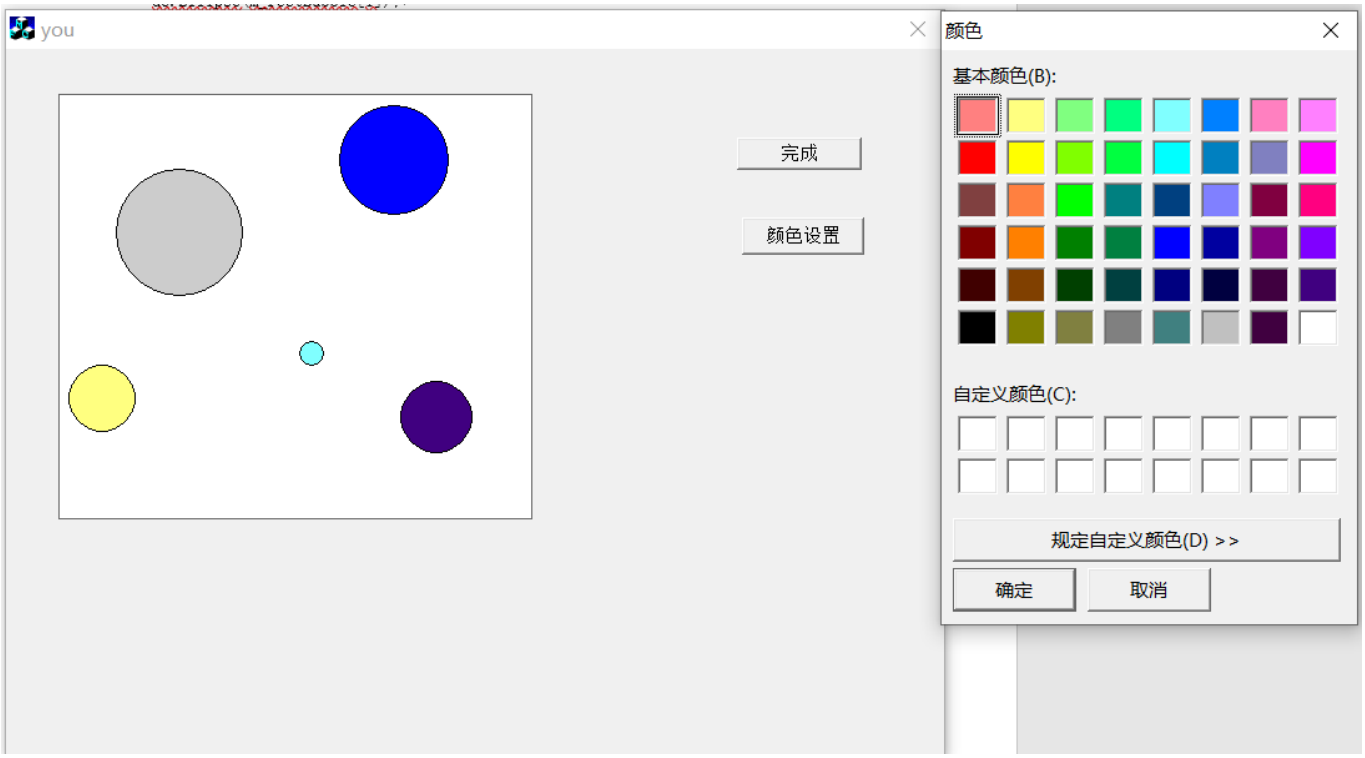
```
            pSysMenu->AppendMenu(MF_STRING, IDM_ABOUTBOX, strAboutMenu);
        }
    }

    // Set the icon for this dialog.  The framework does this automatically
    //  when the application's main window is not a dialog
    SetIcon(m_hIcon, TRUE);            // Set big icon
    SetIcon(m_hIcon, FALSE);       // Set small icon

    // TODO: Add extra initialization here
    CStatic *pST=(CStatic *)GetDlgItem(IDC_CLIENT);
    pST->GetWindowRect(&m_rectClient);
    ScreenToClient(&m_rectClient);
    m_nBubbleCount=0;
    return TRUE;  // return TRUE  unless you set the focus to a control
}
```

//修改 OnPaint()成员函数

```
void CYouDlg::OnPaint()
{
    if (IsIconic())
    {
        CPaintDC dc(this); // device context for painting

        SendMessage(WM_ICONERASEBKGND, (WPARAM) dc.GetSafeHdc(), 0);

        // Center icon in client rectangle
        int cxIcon = GetSystemMetrics(SM_CXICON);
        int cyIcon = GetSystemMetrics(SM_CYICON);
        CRect rect;
        GetClientRect(&rect);
        int x = (rect.Width() - cxIcon + 1) / 2;
        int y = (rect.Height() - cyIcon + 1) / 2;

        // Draw the icon
        dc.DrawIcon(x, y, m_hIcon);
    }
    else
    {CPaintDC dc(this);
    CRgn rgn;
    rgn.CreateRectRgnIndirect(&m_rectClient);
    dc.SelectClipRgn(&rgn);
    dc.Rectangle(m_rectClient);
    CBrush brushNew, *pbrushOld;
    for(int i=0;i<m_nBubbleCount;i++)
```

```
        {
            brushNew.CreateSolidBrush(m_colorBubble[i]);
            pbrushOld=dc.SelectObject(&brushNew);
            dc.Ellipse(m_rectBubble[i]);
            dc.SelectObject(pbrushOld);
            brushNew.DeleteObject();
        }
            CDialog::OnPaint();
        }
}
//修改鼠标左键消息响应函数 OnLButtonDown()
void CYouDlg::OnLButtonDown(UINT nFlags, CPoint point)
{
    // TODO: Add your message handler code here and/or call default
    if(m_nBubbleCount<MAX_BUBBLE)
    {
        int r=rand()%50+10;
        CRect rect(point.x-r,point.y-r,point.x+r,point.y+r);
        m_rectBubble[m_nBubbleCount]=rect;
        m_colorBubble[m_nBubbleCount]=m_colorCurrent;
        m_nBubbleCount++;
        InvalidateRect(rect,FALSE);
    }
    CDialog::OnLButtonDown(nFlags, point);
}
//修改 OnColor()函数
void CYouDlg::OnColor()
{
    // TODO: Add your control notification handler code here
    m_colorCurrent=RGB(200,200,200);
    CColorDialog dlg(m_colorCurrent);
    if(dlg.DoModal()==IDOK)
        m_colorCurrent=dlg.GetColor();
}
```

运行结果：



3. 计算器

题目要求：编写一个计算器程序。该计算器程序使用编辑控件直接输入数据，并有加，减，乘，除，平方根和倒数计算功能。

程序说明：建立一个基于对话框的应用程序框架。编辑对话框模板资源，删除静态文本和 Cancel 按钮，添加一个编辑控件，将其 ID 改为 **IDC_INPUT**，设置 Align text 属性为 **right**。添加 8 个按钮，将其 ID 和名字分别改成：

ID	Caption
IDC_ADD	+
IDC_SUB	-
IDC_MUL	*
IDC_DIV	/
IDC_CLEAR	C
IDC_SQRT	SQRT
IDC_CALC	=
IDC_RECIPROCAL	1/X

修改对话框的模板和以上各控件的尺寸，位置，再利用类向导为编辑控件 IDC_INPUT 添加一个映射函数 **OnSetfocusInput()**。最后再次用类向导为新添的 8 个按钮添加鼠标单击命令函数：**OnAdd(), OnSub(), OnMul(), OnDiv(), OnClear(), OnCalc(), OnReciprocal()和 OnSqrt()**。

模块组成：编辑控件，基本对话框，按钮控件

程序分模块实现：

```
//在对话框类中添加数据成员
class CAaaaaaaDlg : public CDialog
{
// Construction
public:
    void Calc();
    CAaaaaaaDlg(CWnd* pParent = NULL);  // standard constructor
    int m_nOP;
    double m_fResult;
   // BOOL UpdataData(BOOL bSaveAndValidate=TRUE);
// Dialog Data
```

```
    //{{AFX_DATA(CAaaaaaaDlg)
    enum { IDD = IDD_AAAAAAA_DIALOG };
    double   m_fInput;
    //}}AFX_DATA
    // ClassWizard generated virtual function overrides
    //{{AFX_VIRTUAL(CAaaaaaaDlg)
    protected:
    virtual void DoDataExchange(CDataExchange* pDX); // DDX/DDV support
    //}}AFX_VIRTUAL
// Implementation
protected:
    HICON m_hIcon;

    // Generated message map functions
    //{{AFX_MSG(CAaaaaaaDlg)
    virtual BOOL OnInitDialog();
    afx_msg void OnSysCommand(UINT nID, LPARAM lParam);
    afx_msg void OnPaint();
    afx_msg HCURSOR OnQueryDragIcon();
    afx_msg void OnSetfocusInput();
    afx_msg void OnAdd();
    afx_msg void OnCalc();
    afx_msg void OnClear();
    afx_msg void OnDiv();
    afx_msg void OnMul();
    afx_msg void OnReciprocal();
    afx_msg void OnSqrt();
    afx_msg void OnSub();
    //}}AFX_MSG
    DECLARE_MESSAGE_MAP()
};
//在 OnInitDialog()函数中进行初始化
CAaaaaaaDlg::CAaaaaaaDlg(CWnd* pParent /*=NULL*/)
    : CDialog(CAaaaaaaDlg::IDD, pParent)
{
    //{{AFX_DATA_INIT(CAaaaaaaDlg)
    m_fInput = 0.0;
    //m_Nop=0;
    //}}AFX_DATA_INIT
    // Note that LoadIcon does not require a subsequent DestroyIcon in Win32
    m_hIcon = AfxGetApp()->LoadIcon(IDR_MAINFRAME);
}
//为对话框类添加 Calc（）成员函数
void CAaaaaaaDlg::Calc()
```

```
{
UpdateData(TRUE);
switch(m_nOP)
{
    case 0:
     m_fResult=m_fInput;
     break;
     case 1:
     m_fResult+=m_fInput;
     break;
     case 2:
     m_fResult-=m_fInput;
     break;
     case 3:
     m_fResult*=m_fInput;
     break;
     case 4:
     m_fResult/=m_fInput;
     break;
     case 5:
     m_fResult=1/m_fInput;
     break;
     case 6:
     m_fResult=sqrt(m_fInput);
     break;
}
m_fInput=m_fResult;
UpdateData(FALSE);
}
//为所有按钮的消息响应函数添加代码
void CAaaaaaaDlg::OnAdd()
{
     // TODO: Add your control notification handler code here
     Calc();
     m_nOP=1;
}

void CAaaaaaaDlg::OnCalc()
{
     // TODO: Add your control notification handler code here
    Calc();
     m_nOP=0;
}
```

```
void CAaaaaaaDlg::OnClear()
{
    // TODO: Add your control notification handler code here
    m_fResult=0.0;
    m_fInput=0.0;
    m_nOP=0;
    UpdateData(FALSE);
}

void CAaaaaaaDlg::OnDiv()
{
    // TODO: Add your control notification handler code here
    Calc();
m_nOP=4;
}

void CAaaaaaaDlg::OnMul()
{
    // TODO: Add your control notification handler code here
    Calc();
    m_nOP=3;
}

void CAaaaaaaDlg::OnReciprocal()
{
    // TODO: Add your control notification handler code here
    m_nOP=5;
    Calc();
    m_nOP=0;
}

void CAaaaaaaDlg::OnSqrt()
{
    // TODO: Add your control notification handler code here
    m_nOP=6;
    Calc();
    m_nOP=0;
}
void CAaaaaaaDlg::OnSub()
{
    // TODO: Add your control notification handler code here
    Calc();
    m_nOP=2;
}
```

运行结果：