 2226 lines coding

// GAMESIZE.cpp : implementation file

//

#include "stdafx.h"

#include "SaoLei.h"

#include "GAMESIZE.h"

#ifdef \_DEBUG

#define new DEBUG\_NEW

#undef THIS\_FILE

static char THIS\_FILE[] = \_\_FILE\_\_;

#endif

/////////////////////////////////////////////////////////////////////////////

// GAMESIZE dialog

GAMESIZE a;

GAMESIZE::GAMESIZE(CWnd\* pParent /\*=NULL\*/)

: CDialog(GAMESIZE::IDD, pParent)

{

//{{AFX\_DATA\_INIT(GAMESIZE)

m\_length = 10;

m\_number = 10;

m\_width = 10;

//}}AFX\_DATA\_INIT

}

void GAMESIZE::DoDataExchange(CDataExchange\* pDX)

{

CDialog::DoDataExchange(pDX);

//{{AFX\_DATA\_MAP(GAMESIZE)

DDX\_Text(pDX, IDC\_L, m\_length);

DDX\_Text(pDX, IDC\_N, m\_number);

DDX\_Text(pDX, IDC\_W, m\_width);

a.m\_length = m\_length;

a.m\_width = m\_width;

a.m\_number = m\_number;

//}}AFX\_DATA\_MAP

}

BEGIN\_MESSAGE\_MAP(GAMESIZE, CDialog)

//{{AFX\_MSG\_MAP(GAMESIZE)

// NOTE: the ClassWizard will add message map macros here

//}}AFX\_MSG\_MAP

END\_MESSAGE\_MAP()

/////////////////////////////////////////////////////////////////////////////

// GAMESIZE message handlers

int GAMESIZE::DoModal()

{

// TODO: Add your specialized code here and/or call the base class

return CDialog::DoModal();

}

// HIGHSCORE.cpp : implementation file

//

#include "stdafx.h"

#include "SaoLei.h"

#include "HIGHSCORE.h"

#include "SETNAME.h"

#include <fstream>

#ifdef \_DEBUG

#define new DEBUG\_NEW

#undef THIS\_FILE

static char THIS\_FILE[] = \_\_FILE\_\_;

#endif

using namespace std;

/////////////////////////////////////////////////////////////////////////////

// CHIGHSCORE dialog

extern SETNAME b;

extern int game\_time;

extern int win;

CHIGHSCORE::CHIGHSCORE(CWnd\* pParent /\*=NULL\*/) //初始化

: CDialog(CHIGHSCORE::IDD, pParent)

{

int x;

FILE \*fp;

fp = fopen("highscore.txt","r"); //读取文件出现问题

//{{AFX\_DATA\_INIT(CHIGHSCORE)

//}}AFX\_DATA\_INIT

for(x=0;x<3;x++) //读取文件数据

{

fscanf(fp,"%s",&p[x]);

fscanf(fp,"%d",&q[x]);

}

for(x=0;x<3;x++)

if( (game\_time<q[x]) && (win==1) )

{

Sort(x);

q[x]=game\_time;

strcpy(p[x],b.m\_name);

break;

}

m\_first\_name = p[0];

m\_second\_name = p[1];

m\_third\_name = p[2];

m\_first\_time = q[0];

m\_second\_time = q[1];

m\_third\_time = q[2];

fp = fopen("highscore.txt","w");

for(x=0;x<3;x++)

{

fprintf(fp,"%s ",p[x]);

fprintf(fp,"%d ",q[x]);

}

fclose(fp);

}

void CHIGHSCORE::DoDataExchange(CDataExchange\* pDX)

{

CDialog::DoDataExchange(pDX);

//{{AFX\_DATA\_MAP(CHIGHSCORE)

DDX\_Text(pDX, IDC\_FIRST\_TIME, m\_first\_time);

DDX\_Text(pDX, IDC\_FIRST\_NAME, m\_first\_name);

DDX\_Text(pDX, IDC\_SECOND\_NAME, m\_second\_name);

DDX\_Text(pDX, IDC\_SECOND\_TIME, m\_second\_time);

DDX\_Text(pDX, IDC\_THIRD\_NAME, m\_third\_name);

DDX\_Text(pDX, IDC\_THIRD\_TIME, m\_third\_time);

//}}AFX\_DATA\_MAP

}

BEGIN\_MESSAGE\_MAP(CHIGHSCORE, CDialog)

//{{AFX\_MSG\_MAP(CHIGHSCORE)

ON\_COMMAND(AFX\_ID\_GAOFEN, OnIdGaofen)

ON\_WM\_LBUTTONDBLCLK()

//}}AFX\_MSG\_MAP

END\_MESSAGE\_MAP()

/////////////////////////////////////////////////////////////////////////////

// CHIGHSCORE message handlers

BOOL CHIGHSCORE::OnCommand(WPARAM wParam, LPARAM lParam)

{

// TODO: Add your specialized code here and/or call the base class

return CDialog::OnCommand(wParam, lParam);

}

int CHIGHSCORE::DoModal()

{

// TODO: Add your specialized code here and/or call the base class

return CDialog::DoModal();

}

void CHIGHSCORE::OnIdGaofen()

{

// TODO: Add your command handler code here

}

void CHIGHSCORE::Sort(int rank) //排序

{

for(int x=2;x>rank;x--)

{

q[x]=q[x-1];

for(int y=0;y<20;y++)

p[x][y]=p[x-1][y];

}

}

// MainFrm.cpp : implementation of the CMainFrame class

//

#include "stdafx.h"

#include "SaoLei.h"

#include "MainFrm.h"

#ifdef \_DEBUG

#define new DEBUG\_NEW

#undef THIS\_FILE

static char THIS\_FILE[] = \_\_FILE\_\_;

#endif

/////////////////////////////////////////////////////////////////////////////

// CMainFrame

IMPLEMENT\_DYNCREATE(CMainFrame, CFrameWnd)

BEGIN\_MESSAGE\_MAP(CMainFrame, CFrameWnd)

//{{AFX\_MSG\_MAP(CMainFrame)

// NOTE - the ClassWizard will add and remove mapping macros here.

// DO NOT EDIT what you see in these blocks of generated code !

ON\_WM\_CREATE()

//}}AFX\_MSG\_MAP

END\_MESSAGE\_MAP()

static UINT indicators[] =

{

ID\_SEPARATOR, // status line indicator

ID\_INDICATOR\_CAPS,

ID\_INDICATOR\_NUM,

ID\_INDICATOR\_SCRL,

};

/////////////////////////////////////////////////////////////////////////////

// CMainFrame construction/destruction

CMainFrame::CMainFrame()

{

// TODO: add member initialization code here

}

CMainFrame::~CMainFrame()

{

}

int CMainFrame::OnCreate(LPCREATESTRUCT lpCreateStruct)

{

if (CFrameWnd::OnCreate(lpCreateStruct) == -1)

return -1;

return 0;

}

BOOL CMainFrame::PreCreateWindow(CREATESTRUCT& cs)

{

if( !CFrameWnd::PreCreateWindow(cs) )

return FALSE;

// TODO: Modify the Window class or styles here by modifying

// the CREATESTRUCT cs

cs.cx=300; //控制窗口大小

cs.cy=350;

return TRUE;

}

/////////////////////////////////////////////////////////////////////////////

// CMainFrame diagnostics

#ifdef \_DEBUG

void CMainFrame::AssertValid() const

{

CFrameWnd::AssertValid();

}

void CMainFrame::Dump(CDumpContext& dc) const

{

CFrameWnd::Dump(dc);

}

#endif //\_DEBUG

/////////////////////////////////////////////////////////////////////////////

// CMainFrame message handlers

// SaoLei.cpp : Defines the class behaviors for the application.

//

#include "stdafx.h"

#include "SaoLei.h"

#include "MainFrm.h"

#include "SaoLeiDoc.h"

#include "SaoLeiView.h"

#include "GAMESIZE.h"

#include "SETNAME.h"

#ifdef \_DEBUG

#define new DEBUG\_NEW

#undef THIS\_FILE

static char THIS\_FILE[] = \_\_FILE\_\_;

#endif

extern int win; //引用游戏结果

/////////////////////////////////////////////////////////////////////////////

// CSaoLeiApp

BEGIN\_MESSAGE\_MAP(CSaoLeiApp, CWinApp)

//{{AFX\_MSG\_MAP(CSaoLeiApp)

ON\_COMMAND(ID\_APP\_ABOUT, OnAppAbout)

// NOTE - the ClassWizard will add and remove mapping macros here.

// DO NOT EDIT what you see in these blocks of generated code!

//}}AFX\_MSG\_MAP

// Standard file based document commands

ON\_COMMAND(ID\_FILE\_NEW, CWinApp::OnFileNew)

ON\_COMMAND(ID\_FILE\_OPEN, CWinApp::OnFileOpen)

// Standard print setup command

ON\_COMMAND(ID\_FILE\_PRINT\_SETUP, CWinApp::OnFilePrintSetup)

END\_MESSAGE\_MAP()

/////////////////////////////////////////////////////////////////////////////

// CSaoLeiApp construction

CSaoLeiApp::CSaoLeiApp()

{

// TODO: add construction code here,

// Place all significant initialization in InitInstance

}

/////////////////////////////////////////////////////////////////////////////

// The one and only CSaoLeiApp object

CSaoLeiApp theApp;

/////////////////////////////////////////////////////////////////////////////

// CSaoLeiApp initialization

BOOL CSaoLeiApp::InitInstance()

{

AfxEnableControlContainer();

GAMESIZE a;

a.DoModal();

// Standard initialization

// If you are not using these features and wish to reduce the size

// of your final executable, you should remove from the following

// the specific initialization routines you do not need.

if(1)

{

SETNAME b;

b.DoModal();

}

#ifdef \_AFXDLL

Enable3dControls(); // Call this when using MFC in a shared DLL

#else

Enable3dControlsStatic(); // Call this when linking to MFC statically

#endif

// Change the registry key under which our settings are stored.

// TODO: You should modify this string to be something appropriate

// such as the name of your company or organization.

SetRegistryKey(\_T("Local AppWizard-Generated Applications"));

LoadStdProfileSettings(); // Load standard INI file options (including MRU)

// Register the application's document templates. Document templates

// serve as the connection between documents, frame windows and views.

CSingleDocTemplate\* pDocTemplate;

pDocTemplate = new CSingleDocTemplate(

IDR\_MAINFRAME,

RUNTIME\_CLASS(CSaoLeiDoc),

RUNTIME\_CLASS(CMainFrame), // main SDI frame window

RUNTIME\_CLASS(CSaoLeiView));

AddDocTemplate(pDocTemplate);

// Parse command line for standard shell commands, DDE, file open

CCommandLineInfo cmdInfo;

ParseCommandLine(cmdInfo);

// Dispatch commands specified on the command line

if (!ProcessShellCommand(cmdInfo))

return FALSE;

// The one and only window has been initialized, so show and update it.

m\_pMainWnd->ShowWindow(SW\_SHOW);

m\_pMainWnd->UpdateWindow();

return TRUE;

}

/////////////////////////////////////////////////////////////////////////////

// CAboutDlg dialog used for App About

class CAboutDlg : public CDialog

{

public:

CAboutDlg();

// Dialog Data

//{{AFX\_DATA(CAboutDlg)

enum { IDD = IDD\_ABOUTBOX };

//}}AFX\_DATA

// ClassWizard generated virtual function overrides

//{{AFX\_VIRTUAL(CAboutDlg)

protected:

virtual void DoDataExchange(CDataExchange\* pDX); // DDX/DDV support

//}}AFX\_VIRTUAL

// Implementation

protected:

//{{AFX\_MSG(CAboutDlg)

// No message handlers

//}}AFX\_MSG

DECLARE\_MESSAGE\_MAP()

};

CAboutDlg::CAboutDlg() : CDialog(CAboutDlg::IDD)

{

//{{AFX\_DATA\_INIT(CAboutDlg)

//}}AFX\_DATA\_INIT

}

void CAboutDlg::DoDataExchange(CDataExchange\* pDX)

{

CDialog::DoDataExchange(pDX);

//{{AFX\_DATA\_MAP(CAboutDlg)

//}}AFX\_DATA\_MAP

}

BEGIN\_MESSAGE\_MAP(CAboutDlg, CDialog)

//{{AFX\_MSG\_MAP(CAboutDlg)

// No message handlers

//}}AFX\_MSG\_MAP

END\_MESSAGE\_MAP()

// App command to run the dialog

void CSaoLeiApp::OnAppAbout()

{

CAboutDlg aboutDlg;

aboutDlg.DoModal();

}

/////////////////////////////////////////////////////////////////////////////

// CSaoLeiApp message handlers

// SaoLeiDoc.cpp : implementation of the CSaoLeiDoc class

//

#include "stdafx.h"

#include "SaoLei.h"

#include "SaoLeiDoc.h"

#ifdef \_DEBUG

#define new DEBUG\_NEW

#undef THIS\_FILE

static char THIS\_FILE[] = \_\_FILE\_\_;

#endif

/////////////////////////////////////////////////////////////////////////////

// CSaoLeiDoc

IMPLEMENT\_DYNCREATE(CSaoLeiDoc, CDocument)

BEGIN\_MESSAGE\_MAP(CSaoLeiDoc, CDocument)

//{{AFX\_MSG\_MAP(CSaoLeiDoc)

// NOTE - the ClassWizard will add and remove mapping macros here.

// DO NOT EDIT what you see in these blocks of generated code!

//}}AFX\_MSG\_MAP

END\_MESSAGE\_MAP()

/////////////////////////////////////////////////////////////////////////////

// CSaoLeiDoc construction/destruction

CSaoLeiDoc::CSaoLeiDoc()

{

// TODO: add one-time construction code here

}

CSaoLeiDoc::~CSaoLeiDoc()

{

}

BOOL CSaoLeiDoc::OnNewDocument()

{

if (!CDocument::OnNewDocument())

return FALSE;

// TODO: add reinitialization code here

// (SDI documents will reuse this document)

return TRUE;

}

/////////////////////////////////////////////////////////////////////////////

// CSaoLeiDoc serialization

void CSaoLeiDoc::Serialize(CArchive& ar)

{

if (ar.IsStoring())

{

// TODO: add storing code here

}

else

{

// TODO: add loading code here

}

}

/////////////////////////////////////////////////////////////////////////////

// CSaoLeiDoc diagnostics

#ifdef \_DEBUG

void CSaoLeiDoc::AssertValid() const

{

CDocument::AssertValid();

}

void CSaoLeiDoc::Dump(CDumpContext& dc) const

{

CDocument::Dump(dc);

}

#endif //\_DEBUG

/////////////////////////////////////////////////////////////////////////////

// CSaoLeiDoc commands// SaoLeiView.cpp : implementation of the CSaoLeiView class

//

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\

高分榜 √ 开启文件存储 优化对话框(等级)

\\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include "stdafx.h"

#include "SaoLei.h"

#include "SaoLeiDoc.h"

#include "SaoLeiView.h"

#include <ctime>

#include "HIGHSCORE.h" //显示高分榜

#include "GAMESIZE.h" //设置游戏规格

#ifdef \_DEBUG

#define new DEBUG\_NEW

#undef THIS\_FILE

static char THIS\_FILE[] = \_\_FILE\_\_;

#endif

#define originX 40 //原点坐标

#define originY 50 //原点坐标

#define length 20 //单位长

extern GAMESIZE a; //通过对话框控制游戏规格

int \_x,\_y; //记录鼠标点击方格

int set\_mine; //是否布雷

int win; //全局变量 记录是否胜利 传递参数值

int game\_time; //全局变量 记录游戏时间

/////////////////////////////////////////////////////////////////////////////

// CSaoLeiView

IMPLEMENT\_DYNCREATE(CSaoLeiView, CView)

BEGIN\_MESSAGE\_MAP(CSaoLeiView, CView)

//{{AFX\_MSG\_MAP(CSaoLeiView)

ON\_WM\_RBUTTONDOWN()

ON\_WM\_LBUTTONUP()

ON\_WM\_TIMER()

ON\_WM\_CREATE()

ON\_COMMAND(AFX\_ID\_GAOFEN, OnIdGaofen)

ON\_WM\_RBUTTONUP()

ON\_WM\_LBUTTONDOWN()

//}}AFX\_MSG\_MAP

// Standard printing commands

ON\_COMMAND(ID\_FILE\_PRINT, CView::OnFilePrint)

ON\_COMMAND(ID\_FILE\_PRINT\_DIRECT, CView::OnFilePrint)

ON\_COMMAND(ID\_FILE\_PRINT\_PREVIEW, CView::OnFilePrintPreview)

END\_MESSAGE\_MAP()

/////////////////////////////////////////////////////////////////////////////

// CSaoLeiView construction/destruction

CSaoLeiView::CSaoLeiView()

{

set\_mine = 0;

X=a.m\_length;

Y=a.m\_width;

Mine\_number=a.m\_number;

for(int i=0;i<X;i++)

for(int j=0;j<Y;j++)

{

flag[i][j]=0;

Click[i][j]=0;

flagnumber[i][j]=0;

}

Time = 0;

second = 0;

}

CSaoLeiView::~CSaoLeiView()

{}

BOOL CSaoLeiView::PreCreateWindow(CREATESTRUCT& cs)

{

// TODO: Modify the Window class or styles here by modifying

// the CREATESTRUCT cs

return CView::PreCreateWindow(cs);

}

/////////////////////////////////////////////////////////////////////////////

// CSaoLeiView drawing

void CSaoLeiView::SetMine()

{

set\_mine=1;

k=0;

srand((unsigned)time(NULL));

int previous; //之前设置的雷

while(k<=Mine\_number) //产雷

{

lei[k]=rand()%(X\*Y);

for(previous=0;previous<k;previous++)

if(lei[k]==lei[previous])

lei[k]=rand()%(X\*Y);

for(previous=0;previous<k;previous++)

if(lei[k]==lei[previous])

lei[k]=rand()%(X\*Y);

k++;

}

for(i=0;i<X;i++) //初始化无雷区为0

for(j=0;j<Y;j++)

fangge[i][j]=0;

for(k=0;k<Mine\_number;k++)

for(i=0;i<X;i++) //布雷 为1

for(j=0;j<Y;j++)

if( (i\*X+j)==lei[k] )

fangge[i][j]=1;

}

void CSaoLeiView::SetNumber()

{

////////////////////////////////////////////////////////////////////////////////////计算每格中的数字

//左上角

if(fangge[0][0]==0)

switch(fangge[0][1] + fangge[1][1] + fangge[1][0])

{

case 0:{number[0][0]=0;break;}

case 1:{number[0][0]=1;break;}

case 2:{number[0][0]=2;break;}

case 3:{number[0][0]=3;break;}

}

//右上角

if(fangge[X-1][0]==0)

switch(fangge[X-2][0] + fangge[X-2][1] + fangge[X-1][1])

{

case 0:{number[X-1][0]=0;break;}

case 1:{number[X-1][0]=1;break;}

case 2:{number[X-1][0]=2;break;}

case 3:{number[X-1][0]=3;break;}

}

//左下角

if(fangge[0][Y-1]==0)

switch(fangge[0][Y-2] + fangge[1][Y-2] + fangge[1][Y-1])

{

case 0:{number[0][Y-1]=0;break;}

case 1:{number[0][Y-1]=1;break;}

case 2:{number[0][Y-1]=2;break;}

case 3:{number[0][Y-1]=3;break;}

}

//右下角

if(fangge[X-1][Y-1]==0)

switch(fangge[X-2][Y-1] + fangge[X-1][Y-2] + fangge[X-2][Y-2])

{

case 0:{number[X-1][Y-1]=0;break;}

case 1:{number[X-1][Y-1]=1;break;}

case 2:{number[X-1][Y-1]=2;break;}

case 3:{number[X-1][Y-1]=3;break;}

}

//上边

for(i=1,j=0;i<X-1;i++)

if(fangge[i][j]==0)

switch(fangge[i-1][j] + fangge[i-1][j+1] + fangge[i][j+1] + fangge[i+1][j] + fangge[i+1][j+1] )

{

case 0:{number[i][j]=0;break;}

case 1:{number[i][j]=1;break;}

case 2:{number[i][j]=2;break;}

case 3:{number[i][j]=3;break;}

case 4:{number[i][j]=4;break;}

case 5:{number[i][j]=5;break;}

}

//下边

for(i=1,j=Y-1;i<X-1;i++)

if(fangge[i][j]==0)

switch(fangge[i-1][j] + fangge[i-1][j-1] + fangge[i][j-1] + fangge[i+1][j] + fangge[i+1][j-1] )

{

case 0:{number[i][j]=0;break;}

case 1:{number[i][j]=1;break;}

case 2:{number[i][j]=2;break;}

case 3:{number[i][j]=3;break;}

case 4:{number[i][j]=4;break;}

case 5:{number[i][j]=5;break;}

}

//左边

for(i=0,j=0;j<Y-1;j++)

if(fangge[i][j]==0)

switch(fangge[i][j-1] + fangge[i][j+1] + fangge[i+1][j] + fangge[i+1][j-1] + fangge[i+1][j+1] )

{

case 0:{number[i][j]=0;break;}

case 1:{number[i][j]=1;break;}

case 2:{number[i][j]=2;break;}

case 3:{number[i][j]=3;break;}

case 4:{number[i][j]=4;break;}

case 5:{number[i][j]=5;break;}

}

//右边

for(i=X-1,j=0;j<Y-1;j++)

if(fangge[i][j]==0)

switch(fangge[i][j-1] + fangge[i][j+1] + fangge[i-1][j] + fangge[i-1][j-1] + fangge[i-1][j+1] )

{

case 0:{number[i][j]=0;break;}

case 1:{number[i][j]=1;break;}

case 2:{number[i][j]=2;break;}

case 3:{number[i][j]=3;break;}

case 4:{number[i][j]=4;break;}

case 5:{number[i][j]=5;break;}

}

//中心数字

for(i=1;i<X-1;i++)

for(j=1;j<Y-1;j++)

if(fangge[i][j]==0)

switch(fangge[i-1][j-1] + fangge[i-1][j] +

fangge[i-1][j+1] + fangge[i][j-1] + fangge[i][j+1] +

fangge[i+1][j] + fangge[i+1][j-1] + fangge[i+1][j+1])

{

case 0:{number[i][j]=0;break;}

case 1:{number[i][j]=1;break;}

case 2:{number[i][j]=2;break;}

case 3:{number[i][j]=3;break;}

case 4:{number[i][j]=4;break;}

case 5:{number[i][j]=5;break;}

case 6:{number[i][j]=6;break;}

case 7:{number[i][j]=7;break;}

case 8:{number[i][j]=8;break;}

}

}

void CSaoLeiView::SetFlagNumber()

{

//左上角

switch(flag[0][1] + flag[1][1] + flag[1][0])

{

case 0:{flagnumber[0][0]=0;break;}

case 1:{flagnumber[0][0]=1;break;}

case 2:{flagnumber[0][0]=2;break;}

case 3:{flagnumber[0][0]=3;break;}

}

//右上角

switch(flag[X-2][0] + flag[X-2][1] + flag[X-1][1])

{

case 0:{flagnumber[X-1][0]=0;break;}

case 1:{flagnumber[X-1][0]=1;break;}

case 2:{flagnumber[X-1][0]=2;break;}

case 3:{flagnumber[X-1][0]=3;break;}

}

//左下角

switch(flag[0][Y-2] + flag[1][Y-2] + flag[1][Y-1])

{

case 0:{flagnumber[0][Y-1]=0;break;}

case 1:{flagnumber[0][Y-1]=1;break;}

case 2:{flagnumber[0][Y-1]=2;break;}

case 3:{flagnumber[0][Y-1]=3;break;}

}

//右下角

switch(flag[X-2][Y-1] + flag[X-1][Y-2] + flag[X-2][Y-2])

{

case 0:{flagnumber[X-1][Y-1]=0;break;}

case 1:{flagnumber[X-1][Y-1]=1;break;}

case 2:{flagnumber[X-1][Y-1]=2;break;}

case 3:{flagnumber[X-1][Y-1]=3;break;}

}

//上边

for(i=1,j=0;i<X-1;i++)

switch(flag[i-1][j] + flag[i-1][j+1] + flag[i][j+1] + flag[i+1][j] + flag[i+1][j+1] )

{

case 0:{flagnumber[i][j]=0;break;}

case 1:{flagnumber[i][j]=1;break;}

case 2:{flagnumber[i][j]=2;break;}

case 3:{flagnumber[i][j]=3;break;}

case 4:{flagnumber[i][j]=4;break;}

case 5:{flagnumber[i][j]=5;break;}

}

//下边

for(i=1,j=Y-1;i<X-1;i++)

switch(flag[i-1][j] + flag[i-1][j-1] + flag[i][j-1] + flag[i+1][j] + flag[i+1][j-1] )

{

case 0:{flagnumber[i][j]=0;break;}

case 1:{flagnumber[i][j]=1;break;}

case 2:{flagnumber[i][j]=2;break;}

case 3:{flagnumber[i][j]=3;break;}

case 4:{flagnumber[i][j]=4;break;}

case 5:{flagnumber[i][j]=5;break;}

}

//左边

for(i=0,j=0;j<Y-1;j++)

switch(flag[i][j-1] + flag[i][j+1] + flag[i+1][j] + flag[i+1][j-1] + flag[i+1][j+1] )

{

case 0:{flagnumber[i][j]=0;break;}

case 1:{flagnumber[i][j]=1;break;}

case 2:{flagnumber[i][j]=2;break;}

case 3:{flagnumber[i][j]=3;break;}

case 4:{flagnumber[i][j]=4;break;}

case 5:{flagnumber[i][j]=5;break;}

}

//右边

for(i=X-1,j=0;j<Y-1;j++)

switch(flag[i][j-1] + flag[i][j+1] + flag[i-1][j] + flag[i-1][j-1] + flag[i-1][j+1] )

{

case 0:{flagnumber[i][j]=0;break;}

case 1:{flagnumber[i][j]=1;break;}

case 2:{flagnumber[i][j]=2;break;}

case 3:{flagnumber[i][j]=3;break;}

case 4:{flagnumber[i][j]=4;break;}

case 5:{flagnumber[i][j]=5;break;}

}

//中心数字

for(i=1;i<X-1;i++)

for(j=1;j<Y-1;j++)

switch(flag[i-1][j-1] + flag[i-1][j] +

flag[i-1][j+1] + flag[i][j-1] + flag[i][j+1] +

flag[i+1][j] + flag[i+1][j-1] + flag[i+1][j+1])

{

case 0:{flagnumber[i][j]=0;break;}

case 1:{flagnumber[i][j]=1;break;}

case 2:{flagnumber[i][j]=2;break;}

case 3:{flagnumber[i][j]=3;break;}

case 4:{flagnumber[i][j]=4;break;}

case 5:{flagnumber[i][j]=5;break;}

case 6:{flagnumber[i][j]=6;break;}

case 7:{flagnumber[i][j]=7;break;}

case 8:{flagnumber[i][j]=8;break;}

}

}

void CSaoLeiView::SetClick()

{

for(int u=0;u<X\*Y;u++) //遍历每个方格

for(int p=0;p<X;p++)

for(int q=0;q<Y;q++)

if( (number[p][q]==0)&& (Click[p][q]==1))

{

if( (p-1>=0) && (p-1<=X-1) && (q-1>=0) && (q-1<=Y-1) && (fangge[p-1][q-1]==0) )

Click[p-1][q-1]=1;

if( (p-1>=0) && (p-1<=X-1) && (q>=0) && (q<=Y-1) && (fangge[p-1][q]==0) )

Click[p-1][q]=1;

if( (p-1>=0) && (p-1<=X-1) && (q+1>=0) && (q+1<=Y-1) && (fangge[p-1][q+1]==0) )

Click[p-1][q+1]=1;

if( (p>=0) && (p<=X-1) && (q-1>=0) && (q-1<=Y-1) && (fangge[p][q-1]==0) )

Click[p][q-1]=1;

if( (p>=0) && (p<=X-1) && (q+1>=0) && (q+1<=Y-1) && (fangge[p][q+1]==0) )

Click[p][q+1]=1;

if( (p+1>=0) && (p+1<=X-1) && (q-1>=0) && (q-1<=Y-1) && (fangge[p+1][q-1]==0) )

Click[p+1][q-1]=1;

if( (p+1>=0) && (p+1<=X-1) && (q>=0) && (q<=Y-1) && (fangge[p+1][q]==0) )

Click[p+1][q]=1;

if( (p+1>=0) && (p+1<=X-1) && (q+1>=0) && (q+1<=Y-1) && (fangge[p+1][q+1]==0) )

Click[p+1][q+1]=1;

}

for(int p=0;p<X;p++)

for(int q=0;q<Y;q++)

{

if( (LeftClick[p][q]==1) && (RightClick[p][q]==1) )

if(number[p][q]==flagnumber[p][q])

{

if( (p-1>=0) && (p-1<=X-1) && (q-1>=0) && (q-1<=Y-1) && (flag[p-1][q-1]==0) )

Click[p-1][q-1]=1;

if( (p-1>=0) && (p-1<=X-1) && (q>=0) && (q<=Y-1) && (flag[p-1][q]==0) )

Click[p-1][q]=1;

if( (p-1>=0) && (p-1<=X-1) && (q+1>=0) && (q+1<=Y-1) && (flag[p-1][q+1]==0) )

Click[p-1][q+1]=1;

if( (p>=0) && (p<=X-1) && (q-1>=0) && (q-1<=Y-1) && (flag[p][q-1]==0) )

Click[p][q-1]=1;

if( (p>=0) && (p<=X-1) && (q+1>=0) && (q+1<=Y-1) && (flag[p][q+1]==0) )

Click[p][q+1]=1;

if( (p+1>=0) && (p+1<=X-1) && (q-1>=0) && (q-1<=Y-1) && (flag[p+1][q-1]==0) )

Click[p+1][q-1]=1;

if( (p+1>=0) && (p+1<=X-1) && (q>=0) && (q<=Y-1) && (flag[p+1][q]==0) )

Click[p+1][q]=1;

if( (p+1>=0) && (p+1<=X-1) && (q+1>=0) && (q+1<=Y-1) && (flag[p+1][q+1]==0) )

Click[p+1][q+1]=1;

}

}

}

void CSaoLeiView::OnDraw(CDC\* pDC)

{

if(set\_mine!=1)

SetMine(); //进行布雷

SetNumber(); //计算数字

SetFlagNumber(); //计算周围红旗数

SetClick(); //点击操作结算

CSaoLeiDoc\* pDoc = GetDocument();

ASSERT\_VALID(pDoc);

// TODO: add draw code for native data here

if(!pDoc)

return ;

CRect HuaXian;

this->GetClientRect(&HuaXian);

int x,y;

for(x=originX; x<(X+1)\*length+originX; x+=length) //画方格

for(y=originY; y<(Y+1)\*length+originY; y+=length)

{

pDC->MoveTo(x,50);

pDC->LineTo(x,y);

pDC->MoveTo(50,y);

pDC->LineTo(x,y);

}

for(i=0;i<X;i++) //输出画布

for(j=0;j<Y;j++)

{

if(Click[i][j]==0)

{

if(flag[i][j]==1)

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITFLAG); //加载位图资源

BITMAP bmp;

bitmap.GetBitmap(&bmp); //从位图中获取信息，并填充BITMAP结构

int w=bmp.bmWidth; //宽

int h=bmp.bmHeight; //高

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap); //构造一个与指定设备兼容的内存设备环境（暂存区），并将位图装入该环境

pDC->StretchBlt(i\*length+originX+1,j\*length+originY+1,length-2,length-2,&dc,0,0,w,h,SRCCOPY);

}

if(flag[i][j]==0)

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_UNCLICKED);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(i\*length+originX+1,j\*length+originY+1,length-2,length-2,&dc,0,0,w,h,SRCCOPY);

}

}

if(Click[i][j]==1)

{

if(fangge[i][j]==0)

switch(number[i][j])

{

case 0:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_0);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(i\*20+originX+1,j\*20+originY+1,length-2,length-2,&dc,0,0,w,h,SRCCOPY);

break;

}

case 1:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_1);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(i\*20+originX+1,j\*20+originY+1,length-2,length-2,&dc,0,0,w,h,SRCCOPY);

break;

}

case 2:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_2);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(i\*20+originX+1,j\*20+originY+1,length-2,length-2,&dc,0,0,w,h,SRCCOPY);

break;

}

case 3:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_3);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(i\*20+originX+1,j\*20+originY+1,length-2,length-2,&dc,0,0,w,h,SRCCOPY);

break;

}

case 4:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_4);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(i\*20+originX+1,j\*20+originY+1,length-2,length-2,&dc,0,0,w,h,SRCCOPY);

break;

}

case 5:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_5);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(i\*20+originX+1,j\*20+originY+1,length-2,length-2,&dc,0,0,w,h,SRCCOPY);

break;

}

case 6:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_6);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(i\*20+originX+1,j\*20+originY+1,length-2,length-2,&dc,0,0,w,h,SRCCOPY);

break;

}

case 7:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_7);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(i\*20+originX+1,j\*20+originY+1,length-2,length-2,&dc,0,0,w,h,SRCCOPY);

break;

}

case 8:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_8);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(i\*20+originX+1,j\*20+originY+1,length-2,length-2,&dc,0,0,w,h,SRCCOPY);

break;

}

}

if(fangge[i][j]==1)

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(i\*20+originX+1,j\*20+originY+1,length-2,length-2,&dc,0,0,w,h,SRCCOPY);

}

}

}

for(i=0;i<X;i++)

for(j=0;j<Y;j++)

if( (fangge[i][j]==1) && (Click[i][j]==1) )

win\_or\_lose=-1;

int judge=0;

for(i=0;i<X;i++)

for(j=0;j<Y;j++)

{

if( (fangge[i][j]==0) && (Click[i][j]==1) )

judge++;

}

if(judge==X\*Y-Mine\_number)

win\_or\_lose=1;

if(win\_or\_lose==1) //判断胜利

{

win=1;

game\_time=second;

KillTimer(1);

// MessageBox("胜利！");

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_VICTORY);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(originX,originY,X\*length+1,Y\*length+1,&dc,0,0,w,h,SRCCOPY);

return;

}

else if(win\_or\_lose==-1)

{

KillTimer(1);

// MessageBox("失败！");

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_FAIL);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(originX,originY,X\*length+1,Y\*length+1,&dc,0,0,w,h,SRCCOPY);

}

switch(second%10)

{

case 0:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BIT\_0);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(length\*2+1,1,length,length,&dc,0,0,w,h,SRCCOPY);

break;

}

case 1:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_1);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(length\*2+1,1,length,length,&dc,0,0,w,h,SRCCOPY);

break;

}

case 2:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_2);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(length\*2+1,1,length,length,&dc,0,0,w,h,SRCCOPY);

break;

}

case 3:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_3);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(length\*2+1,1,length,length,&dc,0,0,w,h,SRCCOPY);

break;

}

case 4:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_4);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(length\*2+1,1,length,length,&dc,0,0,w,h,SRCCOPY);

break;

}

case 5:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_5);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(length\*2+1,1,length,length,&dc,0,0,w,h,SRCCOPY);

break;

}

case 6:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_6);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(length\*2+1,1,length,length,&dc,0,0,w,h,SRCCOPY);

break;

}

case 7:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_7);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(length\*2+1,1,length,length,&dc,0,0,w,h,SRCCOPY);

break;

}

case 8:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_8);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(length\*2+1,1,length,length,&dc,0,0,w,h,SRCCOPY);

break;

}

case 9:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_9);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(length\*2+1,1,length,length,&dc,0,0,w,h,SRCCOPY);

break;

}

}

switch( (second%100)/10 )

{

case 0:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BIT\_0);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(length+1,1,length,length,&dc,0,0,w,h,SRCCOPY);

break;

}

case 1:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_1);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(length+1,1,length,length,&dc,0,0,w,h,SRCCOPY);

break;

}

case 2:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_2);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(length+1,1,length,length,&dc,0,0,w,h,SRCCOPY);

break;

}

case 3:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_3);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(length+1,1,length,length,&dc,0,0,w,h,SRCCOPY);

break;

}

case 4:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_4);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(length+1,1,length,length,&dc,0,0,w,h,SRCCOPY);

break;

}

case 5:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_5);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(length+1,1,length,length,&dc,0,0,w,h,SRCCOPY);

break;

}

case 6:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_6);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(length+1,1,length,length,&dc,0,0,w,h,SRCCOPY);

break;

}

case 7:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_7);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(length+1,1,length,length,&dc,0,0,w,h,SRCCOPY);

break;

}

case 8:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_8);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(length+1,1,length,length,&dc,0,0,w,h,SRCCOPY);

break;

}

case 9:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_9);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(length+1,1,length,length,&dc,0,0,w,h,SRCCOPY);

break;

}

}

switch(second/100)

{

case 0:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BIT\_0);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(1,1,length,length,&dc,0,0,w,h,SRCCOPY);

break;

}

case 1:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_1);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(1,1,length,length,&dc,0,0,w,h,SRCCOPY);

break;

}

case 2:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_2);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(1,1,length,length,&dc,0,0,w,h,SRCCOPY);

break;

}

case 3:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_3);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(1,1,length,length,&dc,0,0,w,h,SRCCOPY);

break;

}

case 4:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_4);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(1,1,length,length,&dc,0,0,w,h,SRCCOPY);

break;

}

case 5:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_5);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(1,1,length,length,&dc,0,0,w,h,SRCCOPY);

break;

}

case 6:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_6);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(1,1,length,length,&dc,0,0,w,h,SRCCOPY);

break;

}

case 7:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_7);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(1,1,length,length,&dc,0,0,w,h,SRCCOPY);

break;

}

case 8:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_8);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(1,1,length,length,&dc,0,0,w,h,SRCCOPY);

break;

}

case 9:

{

CBitmap bitmap;

bitmap.LoadBitmap(IDB\_BITMINE\_9);

BITMAP bmp;

bitmap.GetBitmap(&bmp);

int w=bmp.bmWidth;

int h=bmp.bmHeight;

CDC dc;

dc.CreateCompatibleDC(pDC);

dc.SelectObject(&bitmap);

pDC->StretchBlt(1,1,length,length,&dc,0,0,w,h,SRCCOPY);

break;

}

}

}

/////////////////////////////////////////////////////////////////////////////

// CSaoLeiView printing

BOOL CSaoLeiView::OnPreparePrinting(CPrintInfo\* pInfo)

{

// default preparation

return DoPreparePrinting(pInfo);

}

void CSaoLeiView::OnBeginPrinting(CDC\* /\*pDC\*/, CPrintInfo\* /\*pInfo\*/)

{

// TODO: add extra initialization before printing

}

void CSaoLeiView::OnEndPrinting(CDC\* /\*pDC\*/, CPrintInfo\* /\*pInfo\*/)

{

// TODO: add cleanup after printing

}

/////////////////////////////////////////////////////////////////////////////

// CSaoLeiView diagnostics

#ifdef \_DEBUG

void CSaoLeiView::AssertValid() const

{

CView::AssertValid();

}

void CSaoLeiView::Dump(CDumpContext& dc) const

{

CView::Dump(dc);

}

CSaoLeiDoc\* CSaoLeiView::GetDocument() // non-debug version is inline

{

ASSERT(m\_pDocument->IsKindOf(RUNTIME\_CLASS(CSaoLeiDoc)));

return (CSaoLeiDoc\*)m\_pDocument;

}

#endif //\_DEBUG

/////////////////////////////////////////////////////////////////////////////

// CSaoLeiView message handlers

void CSaoLeiView::OnLButtonUp(UINT nFlags, CPoint point)

{

if(Time != 1)

Time = 1; //开始计时

for(int SquareX=0;SquareX<=X;SquareX++) //判断位于哪个方格

for(int SquareY=0;SquareY<=Y;SquareY++)

if( (point.x>=SquareX\*length+originX) && (point.x<= (SquareX+1) \*length+originX) )

if( (point.y>=SquareY\*length+originY) && (point.y<= (SquareY+1) \*length+originY) )

{

\_x=SquareX;

\_y=SquareY;

}

if( (win\_or\_lose!=1) && (win\_or\_lose!=-1) )

if( (\_x<X) && (\_x>=0) && (\_y<Y) && (\_y>=0) )

{

Click[\_x][\_y]=1; //记录：左键被点开

LeftClick[\_x][\_y]=1; //记录：左键摁下

Invalidate();

}

CView::OnLButtonUp(nFlags, point);

}

void CSaoLeiView::OnTimer(UINT nIDEvent)

{

if( (win\_or\_lose==1) || (win\_or\_lose==-1) ) //判断游戏是否结束

return ;

if(Time>0) //开始计时

{

second++;

CRect rect; //控制重画部分

rect.left=1;

rect.right=3\*length+1;

rect.top=1;

rect.bottom=length+1;

InvalidateRect(&rect);

}

CView::OnTimer(nIDEvent);

}

BOOL CSaoLeiView::Create(LPCTSTR lpszClassName, LPCTSTR lpszWindowName, DWORD dwStyle, const RECT& rect, CWnd\* pParentWnd, UINT nID, CCreateContext\* pContext)

{

// TODO: Add your specialized code here and/or call the base class

return CWnd::Create(lpszClassName, lpszWindowName, dwStyle, rect, pParentWnd, nID, pContext);

}

int CSaoLeiView::OnCreate(LPCREATESTRUCT lpCreateStruct)

{

if (CView::OnCreate(lpCreateStruct) == -1)

return -1;

SetTimer(1,1000,NULL); //计时器

return 0;

}

void CSaoLeiView::OnIdGaofen()

{

// TODO: Add your command handler code here

CHIGHSCORE highscore;

highscore.DoModal();

}

void CSaoLeiView::OnRButtonUp(UINT nFlags, CPoint point)

{

// TODO: Add your message handler code here and/or call default

for(int SquareX=0;SquareX<=X;SquareX++) //判断位于哪个方格

for(int SquareY=0;SquareY<=Y;SquareY++)

if( (point.x>=SquareX\*length+originX) && (point.x<= (SquareX+1) \*length+originX) )

if( (point.y>=SquareY\*length+originY) && (point.y<= (SquareY+1) \*length+originY) )

{

\_x=SquareX;

\_y=SquareY;

}

if( (win\_or\_lose!=1) && (win\_or\_lose!=-1) ) //判断是否结束游戏

if( (\_x<X) && (\_x>=0) && (\_y<Y) && (\_y>=0) )

{

if(flag[\_x][\_y]==0)

flag[\_x][\_y]=1; //记录：被标记

else

if(flag[\_x][\_y]==1)

flag[\_x][\_y]=0;

RightClick[\_x][\_y]=1;

Invalidate();

}

CView::OnRButtonUp(nFlags, point);

}// SETNAME.cpp : implementation file

//

#include "stdafx.h"

#include "SaoLei.h"

#include "SETNAME.h"

#ifdef \_DEBUG

#define new DEBUG\_NEW

#undef THIS\_FILE

static char THIS\_FILE[] = \_\_FILE\_\_;

#endif

/////////////////////////////////////////////////////////////////////////////

// SETNAME dialog

SETNAME b;

SETNAME::SETNAME(CWnd\* pParent /\*=NULL\*/)

: CDialog(SETNAME::IDD, pParent)

{

//{{AFX\_DATA\_INIT(SETNAME)

m\_name = \_T("");

//}}AFX\_DATA\_INIT

}

void SETNAME::DoDataExchange(CDataExchange\* pDX)

{

CDialog::DoDataExchange(pDX);

//{{AFX\_DATA\_MAP(SETNAME)

DDX\_Text(pDX, IDC\_NAME, m\_name);

b.m\_name=m\_name;

//}}AFX\_DATA\_MAP

}

BEGIN\_MESSAGE\_MAP(SETNAME, CDialog)

//{{AFX\_MSG\_MAP(SETNAME)

// NOTE: the ClassWizard will add message map macros here

//}}AFX\_MSG\_MAP

END\_MESSAGE\_MAP()

/////////////////////////////////////////////////////////////////////////////

// SETNAME message handlers#if !defined(AFX\_GAMESIZE\_H\_\_552029C6\_8A88\_4BC4\_9FB6\_D3EC17918A42\_\_INCLUDED\_)

#define AFX\_GAMESIZE\_H\_\_552029C6\_8A88\_4BC4\_9FB6\_D3EC17918A42\_\_INCLUDED\_

#if \_MSC\_VER > 1000

#pragma once

#endif // \_MSC\_VER > 1000

// GAMESIZE.h : header file

//

/////////////////////////////////////////////////////////////////////////////

// GAMESIZE dialog

class GAMESIZE : public CDialog

{

// Construction

public:

GAMESIZE(CWnd\* pParent = NULL); // standard constructor

// Dialog Data

//{{AFX\_DATA(GAMESIZE)

enum { IDD = IDD\_SIZE };

int m\_length;

int m\_number;

int m\_width;

//}}AFX\_DATA

// Overrides

// ClassWizard generated virtual function overrides

//{{AFX\_VIRTUAL(GAMESIZE)

public:

virtual int DoModal();

protected:

virtual void DoDataExchange(CDataExchange\* pDX); // DDX/DDV support

//}}AFX\_VIRTUAL

// Implementation

protected:

// Generated message map functions

//{{AFX\_MSG(GAMESIZE)

// NOTE: the ClassWizard will add member functions here

//}}AFX\_MSG

DECLARE\_MESSAGE\_MAP()

};

//{{AFX\_INSERT\_LOCATION}}

// Microsoft Visual C++ will insert additional declarations immediately before the previous line.

#endif // !defined(AFX\_GAMESIZE\_H\_\_552029C6\_8A88\_4BC4\_9FB6\_D3EC17918A42\_\_INCLUDED\_)#if !defined(AFX\_HIGHSCORE\_H\_\_3E1F11B8\_02D7\_4C68\_8393\_D4FEC5DC0D1F\_\_INCLUDED\_)

#define AFX\_HIGHSCORE\_H\_\_3E1F11B8\_02D7\_4C68\_8393\_D4FEC5DC0D1F\_\_INCLUDED\_

#if \_MSC\_VER > 1000

#pragma once

#endif // \_MSC\_VER > 1000

// HIGHSCORE.h : header file

//

/////////////////////////////////////////////////////////////////////////////

// CHIGHSCORE dialog

class CHIGHSCORE : public CDialog

{

// Construction

public:

CHIGHSCORE(CWnd\* pParent = NULL); // standard constructor

void Sort(int rank);

// Dialog Data

//{{AFX\_DATA(CHIGHSCORE)

enum { IDD = IDD\_HIGHSCORE };

int m\_first\_time;

CString m\_first\_name;

CString m\_second\_name;

int m\_second\_time;

CString m\_third\_name;

int m\_third\_time;

//}}AFX\_DATA

char p[3][20];

int q[3];

// Overrides

// ClassWizard generated virtual function overrides

//{{AFX\_VIRTUAL(CHIGHSCORE)

public:

virtual int DoModal();

protected:

virtual void DoDataExchange(CDataExchange\* pDX); // DDX/DDV support

virtual BOOL OnCommand(WPARAM wParam, LPARAM lParam);

//}}AFX\_VIRTUAL

// Implementation

protected:

// Generated message map functions

//{{AFX\_MSG(CHIGHSCORE)

afx\_msg void OnIdGaofen();

//}}AFX\_MSG

DECLARE\_MESSAGE\_MAP()

};

//{{AFX\_INSERT\_LOCATION}}

// Microsoft Visual C++ will insert additional declarations immediately before the previous line.

#endif // !defined(AFX\_HIGHSCORE\_H\_\_3E1F11B8\_02D7\_4C68\_8393\_D4FEC5DC0D1F\_\_INCLUDED\_)// MainFrm.h : interface of the CMainFrame class

//

/////////////////////////////////////////////////////////////////////////////

#if !defined(AFX\_MAINFRM\_H\_\_8A325758\_A621\_4ABC\_BB20\_B43935311FB3\_\_INCLUDED\_)

#define AFX\_MAINFRM\_H\_\_8A325758\_A621\_4ABC\_BB20\_B43935311FB3\_\_INCLUDED\_

#if \_MSC\_VER > 1000

#pragma once

#endif // \_MSC\_VER > 1000

class CMainFrame : public CFrameWnd

{

protected: // create from serialization only

CMainFrame();

DECLARE\_DYNCREATE(CMainFrame)

// Attributes

public:

// Operations

public:

// Overrides

// ClassWizard generated virtual function overrides

//{{AFX\_VIRTUAL(CMainFrame)

virtual BOOL PreCreateWindow(CREATESTRUCT& cs);

//}}AFX\_VIRTUAL

// Implementation

public:

virtual ~CMainFrame();

#ifdef \_DEBUG

virtual void AssertValid() const;

virtual void Dump(CDumpContext& dc) const;

#endif

protected: // control bar embedded members

CStatusBar m\_wndStatusBar;

CToolBar m\_wndToolBar;

// Generated message map functions

protected:

//{{AFX\_MSG(CMainFrame)

afx\_msg int OnCreate(LPCREATESTRUCT lpCreateStruct);

// 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()

};

/////////////////////////////////////////////////////////////////////////////

//{{AFX\_INSERT\_LOCATION}}

// Microsoft Visual C++ will insert additional declarations immediately before the previous line.

#endif // !defined(AFX\_MAINFRM\_H\_\_8A325758\_A621\_4ABC\_BB20\_B43935311FB3\_\_INCLUDED\_)// SaoLei.h : main header file for the SAOLEI application

//

#if !defined(AFX\_SAOLEI\_H\_\_3FD4DA52\_C5DB\_4577\_A51D\_AFBA96D966C3\_\_INCLUDED\_)

#define AFX\_SAOLEI\_H\_\_3FD4DA52\_C5DB\_4577\_A51D\_AFBA96D966C3\_\_INCLUDED\_

#if \_MSC\_VER > 1000

#pragma once

#endif // \_MSC\_VER > 1000

#ifndef \_\_AFXWIN\_H\_\_

#error include 'stdafx.h' before including this file for PCH

#endif

#include "resource.h" // main symbols

/////////////////////////////////////////////////////////////////////////////

// CSaoLeiApp:

// See SaoLei.cpp for the implementation of this class

//

class CSaoLeiApp : public CWinApp

{

public:

CSaoLeiApp();

// Overrides

// ClassWizard generated virtual function overrides

//{{AFX\_VIRTUAL(CSaoLeiApp)

public:

virtual BOOL InitInstance();

//}}AFX\_VIRTUAL

// Implementation

//{{AFX\_MSG(CSaoLeiApp)

afx\_msg void OnAppAbout();

// 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()

};

/////////////////////////////////////////////////////////////////////////////

//{{AFX\_INSERT\_LOCATION}}

// Microsoft Visual C++ will insert additional declarations immediately before the previous line.

#endif // !defined(AFX\_SAOLEI\_H\_\_3FD4DA52\_C5DB\_4577\_A51D\_AFBA96D966C3\_\_INCLUDED\_)// SaoLeiDoc.h : interface of the CSaoLeiDoc class

//

/////////////////////////////////////////////////////////////////////////////

#if !defined(AFX\_SAOLEIDOC\_H\_\_2782C80C\_0C62\_488C\_BE87\_CD15F2AA0CCD\_\_INCLUDED\_)

#define AFX\_SAOLEIDOC\_H\_\_2782C80C\_0C62\_488C\_BE87\_CD15F2AA0CCD\_\_INCLUDED\_

#if \_MSC\_VER > 1000

#pragma once

#endif // \_MSC\_VER > 1000

class CSaoLeiDoc : public CDocument

{

protected: // create from serialization only

CSaoLeiDoc();

DECLARE\_DYNCREATE(CSaoLeiDoc)

// Attributes

public:

// Operations

public:

// Overrides

// ClassWizard generated virtual function overrides

//{{AFX\_VIRTUAL(CSaoLeiDoc)

public:

virtual BOOL OnNewDocument();

virtual void Serialize(CArchive& ar);

//}}AFX\_VIRTUAL

// Implementation

public:

virtual ~CSaoLeiDoc();

#ifdef \_DEBUG

virtual void AssertValid() const;

virtual void Dump(CDumpContext& dc) const;

#endif

protected:

// Generated message map functions

protected:

//{{AFX\_MSG(CSaoLeiDoc)

// 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()

};

/////////////////////////////////////////////////////////////////////////////

//{{AFX\_INSERT\_LOCATION}}

// Microsoft Visual C++ will insert additional declarations immediately before the previous line.

#endif // !defined(AFX\_SAOLEIDOC\_H\_\_2782C80C\_0C62\_488C\_BE87\_CD15F2AA0CCD\_\_INCLUDED\_)// SaoLeiView.h : interface of the CSaoLeiView class

//

/////////////////////////////////////////////////////////////////////////////

#if !defined(AFX\_SAOLEIVIEW\_H\_\_5C27BEF7\_F572\_49F7\_872B\_ED3227BEE0B8\_\_INCLUDED\_)

#define AFX\_SAOLEIVIEW\_H\_\_5C27BEF7\_F572\_49F7\_872B\_ED3227BEE0B8\_\_INCLUDED\_

#if \_MSC\_VER > 1000

#pragma once

#endif // \_MSC\_VER > 1000

class CSaoLeiView : public CView //最大游戏界面 50\*50

{

int fangge[50][50]; //记录是否有雷

int number[50][50]; //记录方框内数字

int Click[50][50]; //记录是否被点开

int flag[50][50]; //记录是否被红旗标记

int LeftClick[50][50];

int RightClick[50][50];

int flagnumber[50][50];

int i,j; //坐标

int k; //雷循环变量

int lei[50]; //产雷上限50

int X; //游戏界面长

int Y; //游戏界面宽

int Mine\_number; //雷数

int win\_or\_lose; //是否胜利(1/-1)

public:

void SetMine(); //布雷函数

void SetNumber(); //计算数字函数

void SetClick(); //点开方格

void SetFlagNumber(); //计算附近红旗数

protected: // create from serialization only

CSaoLeiView();

DECLARE\_DYNCREATE(CSaoLeiView)

// Attributes

public:

CSaoLeiDoc\* GetDocument();

// Operations

public:

int Time; //开始计时标志

int second; //计时

// Overrides

// ClassWizard generated virtual function overrides

//{{AFX\_VIRTUAL(CSaoLeiView)

public:

virtual void OnDraw(CDC\* pDC); // overridden to draw this view

virtual BOOL PreCreateWindow(CREATESTRUCT& cs);

virtual BOOL Create(LPCTSTR lpszClassName, LPCTSTR lpszWindowName, DWORD dwStyle, const RECT& rect, CWnd\* pParentWnd, UINT nID, CCreateContext\* pContext = NULL);

protected:

virtual BOOL OnPreparePrinting(CPrintInfo\* pInfo);

virtual void OnBeginPrinting(CDC\* pDC, CPrintInfo\* pInfo);

virtual void OnEndPrinting(CDC\* pDC, CPrintInfo\* pInfo);

//}}AFX\_VIRTUAL

// Implementation

public:

virtual ~CSaoLeiView();

#ifdef \_DEBUG

virtual void AssertValid() const;

virtual void Dump(CDumpContext& dc) const;

#endif

protected:

// Generated message map functions

protected:

//{{AFX\_MSG(CSaoLeiView)

afx\_msg void OnLButtonUp(UINT nFlags, CPoint point);

afx\_msg void OnTimer(UINT nIDEvent);

afx\_msg int OnCreate(LPCREATESTRUCT lpCreateStruct);

afx\_msg void OnIdGaofen();

afx\_msg void OnRButtonUp(UINT nFlags, CPoint point);

//}}AFX\_MSG

DECLARE\_MESSAGE\_MAP()

};

#ifndef \_DEBUG // debug version in SaoLeiView.cpp

inline CSaoLeiDoc\* CSaoLeiView::GetDocument()

{ return (CSaoLeiDoc\*)m\_pDocument; }

#endif

/////////////////////////////////////////////////////////////////////////////

//{{AFX\_INSERT\_LOCATION}}

// Microsoft Visual C++ will insert additional declarations immediately before the previous line.

#endif // !defined(AFX\_SAOLEIVIEW\_H\_\_5C27BEF7\_F572\_49F7\_872B\_ED3227BEE0B8\_\_INCLUDED\_)#if !defined(AFX\_SETNAME\_H\_\_646748B5\_0793\_4DDD\_93DA\_379688096363\_\_INCLUDED\_)

#define AFX\_SETNAME\_H\_\_646748B5\_0793\_4DDD\_93DA\_379688096363\_\_INCLUDED\_

#if \_MSC\_VER > 1000

#pragma once

#endif // \_MSC\_VER > 1000

// SETNAME.h : header file

//

/////////////////////////////////////////////////////////////////////////////

// SETNAME dialog

class SETNAME : public CDialog

{

// Construction

public:

SETNAME(CWnd\* pParent = NULL); // standard constructor

// Dialog Data

//{{AFX\_DATA(SETNAME)

enum { IDD = IDD\_SETNAME };

CString m\_name;

//}}AFX\_DATA

// Overrides

// ClassWizard generated virtual function overrides

//{{AFX\_VIRTUAL(SETNAME)

protected:

virtual void DoDataExchange(CDataExchange\* pDX); // DDX/DDV support

//}}AFX\_VIRTUAL

// Implementation

protected:

// Generated message map functions

//{{AFX\_MSG(SETNAME)

// NOTE: the ClassWizard will add member functions here

//}}AFX\_MSG

DECLARE\_MESSAGE\_MAP()

};

//{{AFX\_INSERT\_LOCATION}}

// Microsoft Visual C++ will insert additional declarations immediately before the previous line.

#endif // !defined(AFX\_SETNAME\_H\_\_646748B5\_0793\_4DDD\_93DA\_379688096363\_\_INCLUDED\_)// stdafx.h : include file for standard system include files,

// or project specific include files that are used frequently, but

// are changed infrequently

//

#if !defined(AFX\_STDAFX\_H\_\_DC740A77\_B2DE\_4BFF\_A6FB\_EC2C7708293D\_\_INCLUDED\_)

#define AFX\_STDAFX\_H\_\_DC740A77\_B2DE\_4BFF\_A6FB\_EC2C7708293D\_\_INCLUDED\_

#if \_MSC\_VER > 1000

#pragma once

#endif // \_MSC\_VER > 1000

#define VC\_EXTRALEAN // Exclude rarely-used stuff from Windows headers

#include <afxwin.h> // MFC core and standard components

#include <afxext.h> // MFC extensions

#include <afxdisp.h> // MFC Automation classes

#include <afxdtctl.h> // MFC support for Internet Explorer 4 Common Controls

#ifndef \_AFX\_NO\_AFXCMN\_SUPPORT

#include <afxcmn.h> // MFC support for Windows Common Controls

#endif // \_AFX\_NO\_AFXCMN\_SUPPORT

//{{AFX\_INSERT\_LOCATION}}

// Microsoft Visual C++ will insert additional declarations immediately before the previous line.

#endif // !defined(AFX\_STDAFX\_H\_\_DC740A77\_B2DE\_4BFF\_A6FB\_EC2C7708293D\_\_INCLUDED\_)