概念题

1. 请简述图形用户接口(GUI)的概念，并说明 GUI 主要由哪些部分组成。

   图形用户接口是人与计算机进行交互的一种方式。

   由窗口、下拉菜单、对话框及其相应的控制机制构成。

   用户通过鼠标、键盘等输入设备操纵屏幕上的图标和菜单向计算机发出指令；

   计算机用图形输出来反馈操作的结果。

2. 打开对话框一般需要哪几个步骤？有哪些预定义的对话框类？作用分别是什么？

   1) 首先要创建一个对话框类的对象；

      然后通过与控件对应的成员变量为控件提供初始值；

      调用对话框类的成员函数 DoModal，显示对话框；

      对话框关闭后，通过与控件对应的成员变量获取用户在控件中输入的内容。

   2) CFileDialog：文件打开/保存对话框

      CFontDialog：字体选择对话框

      CColorDialog：颜色选择对话框

      CPrintDialog：打印设置对话框

      CFindReplaceDialog：查找/替换对话框

3. 请列举一些 CView 的派生类，并分别说明它们的功能。

   CScrollView（带滚动功能的视）

   CEditView（具有编辑功能的视）

   CFormView（具有表格功能的视）

   CHtmlView（具有 Web 浏览功能的视）

编程题

1. 创建一个 MFC 应用程序，可以选择文字属性并显示文本信息。

   程序有“文件”和“编辑”菜单项，“编辑”菜单项有三个选项，“文件”菜单项有两个选项

```
// CDialogInput.h
class CDialogInput : public CDialogEx
{
......
protected:
	virtual BOOL OnInitDialog();
public:
	CString strText = "";
	CEdit m_edit;
	afx_msg void OnEnChangeEdit();
	afx_msg void OnBnClickedOk();
	afx_msg void OnBnClickedCancel();
};

// CDialogInput.cpp
```

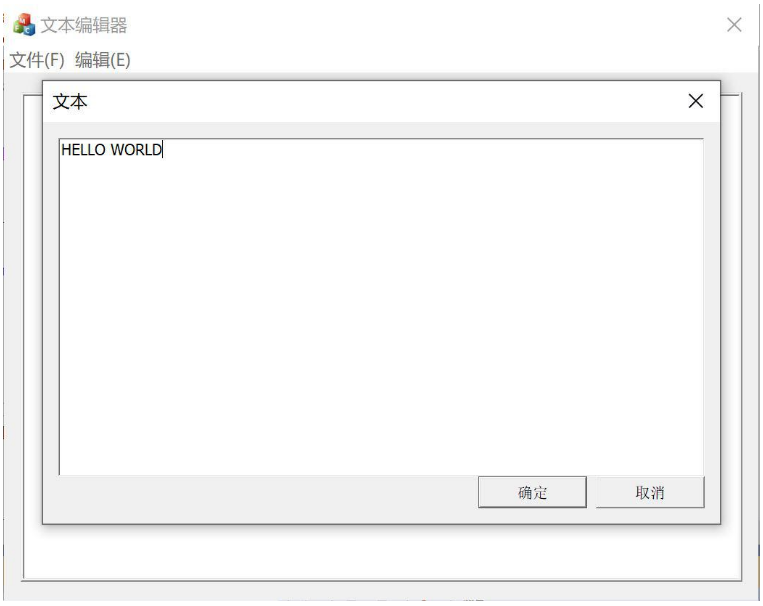
```
BOOL CDialogInput::OnInitDialog()
{
	CDialogEx::OnInitDialog();
	m_edit.SetWindowText(strText);
	return TRUE;
}


void CDialogInput::OnEnChangeEdit()
{
	CString strText;
	m_edit.GetWindowText(strText);
	strText = strText;
}


void CDialogInput::OnBnClickedOk()
{
	CDialogEx::OnOK();
	GetDlgItem(IDC_EDIT)->GetWindowText(strText);
}


void CDialogInput::OnBnClickedCancel()
{
	CDialogEx::OnCancel();
}
```

```
// NotepadDlg.h
class CNotepadDlg : public CDialogEx
{
......
protected:
	HICON m_hIcon;
	CFont m_Font;
public:
	CString strOld = "";
	CString strNew = "";
	CString strOpenPath = "";
	int Save = 0;
	CRichEditCtrl m_FileText;
	afx_msg void OnSave();
	afx_msg void OnEnChangeFile();
	afx_msg void FileSave();
```

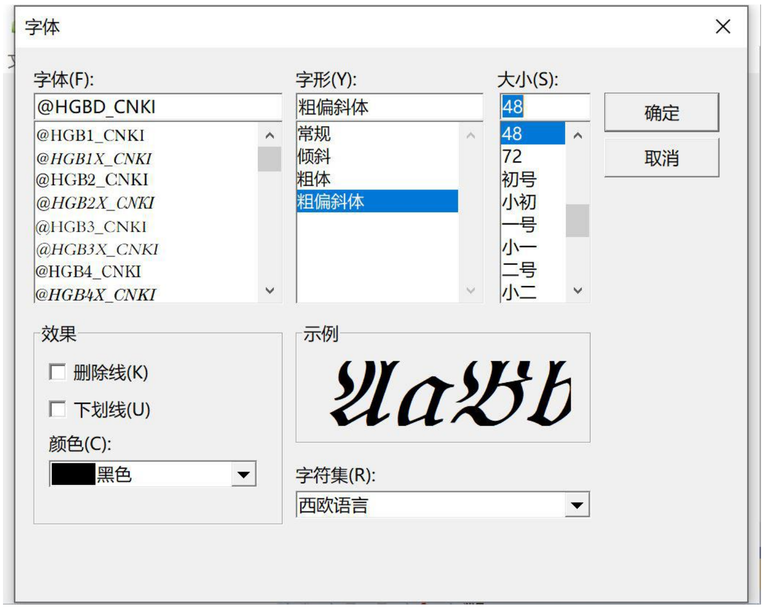
```
    afx_msg void OnOpen();
    afx_msg void OnFont();
    afx_msg void OnColor();
    afx_msg void OnText();
};

void CNotepadDlg::FileSave()
{
    CString   strText = "";
    char write[10000];
    if ((strOpenPath.Right(4) != ".TXT") && (strOpenPath.Right(4) != ".txt"))
        strOpenPath += ".TXT";
    CFile file(_T(strOpenPath), CFile::modeCreate | CFile::modeWrite);
    m_FileText.GetWindowText(strText);
    strcpy(write, strText);
    file.Write(write, strText.GetLength());
    strOld = strNew;
    Save = 1;
    file.Close();
}
```

```
// NotepadDlg.cpp
void CNotepadDlg::OnSave()
{
    if (strOpenPath == "") {
        CFileDialog dlg(FALSE, NULL, NULL, OFN_HIDEREADONLY |
            OFN_OVERWRITEPROMPT, "All Files(*.TXT)|*.TXT||", AfxGetMainWnd());
        CString strPathAs, strText = "";
        char write[10000];
        if (dlg.DoModal() == IDOK) {
            strPathAs = dlg.GetPathName();
            strOpenPath = strPathAs;
            if ((strPathAs.Right(4) != ".TXT") && (strPathAs.Right(4) != ".txt"))
                strPathAs += ".TXT";
            CFile file(_T(strPathAs), CFile::modeCreate | CFile::modeWrite);
            m_FileText.GetWindowText(strText);
            strcpy(write, strText);
            file.Write(write, strText.GetLength());
            strOld = strNew;
            Save = 1;
            file.Close();
        }
        else
            Save = 0;
    }
    else
        FileSave();
}

void CNotepadDlg::OnEnChangeFile()
{
    CString strText;
    m_FileText.GetWindowText(strText);
    strNew = strText;
}

void CNotepadDlg::OnOpen()
{
    if (strOld != strNew) {
        if (MessageBox("内容已改变要保存吗？ ", NULL, MB_YESNO | MB_ICONQUESTION)
== IDYES) {
            OnSave();
            if (Save == 0)    return;
        }
    }
    CFileDialog dlg(TRUE, NULL, NULL, OFN_HIDEREADONLY | OFN_OVERWRITEPROMPT,
        "All Files(*.TXT)|*.TXT||", AfxGetMainWnd());
    CString strText = "";
    if (dlg.DoModal() == IDOK) {
        strOpenPath = dlg.GetPathName();
        CFile file(strOpenPath, CFile::modeRead);
        char read[10000];
        file.Read(read, 10000);
        for (int i = 0; i < file.GetLength(); i++) {
            strText += read[i];
        }
        strOld = strText;
        strNew = strText;
        file.Close();
        m_FileText.SetWindowText(strOld);
    }
}

void CNotepadDlg::OnFont()
{
    CHARFORMAT cf = { 0 };
    cf.cbSize = sizeof(cf);
    m_FileText.GetSelectionCharFormat(cf);
```

```
	CFontDialog dlg(cf);
	if (dlg.DoModal() == IDOK) {
		dlg.GetCharFormat(cf);
		m_FileText.SetSelectionCharFormat(cf);
	}
}

void CNotepadDlg::OnColor()
{
	CHARFORMAT cf = { 0 };
	cf.cbSize = sizeof(cf);
	cf.dwMask = CFM_COLOR;
	m_FileText.GetSelectionCharFormat(cf);
	CColorDialog dlg(cf.crTextColor);
	if (dlg.DoModal() == IDOK) {
		cf.crTextColor = dlg.GetColor();
		m_FileText.SetSelectionCharFormat(cf);
	}
}

void CNotepadDlg::OnText()
{
	CDialogInput dlg;
	dlg.strText = strNew;
	if (dlg.DoModal() == IDOK) {
		strNew = dlg.strText;
		m_FileText.SetWindowText(strNew);
	}
}
```



HELLO WORLD



字体

字体(F): @HGBD_CNKI

@HGB1_CNKI
@HGB1X_CNKI
@HGB2_CNKI
@HGB2X_CNKI
@HGB3_CNKI
@HGB3X_CNKI
@HGB4_CNKI
@HGB4X_CNKI

字形(Y): 粗偏斜体

常规
倾斜
粗体
粗偏斜体

大小(S): 48

48
72
初号
小初
一号
小一
二号
小二

效果

□ 删除线(K)

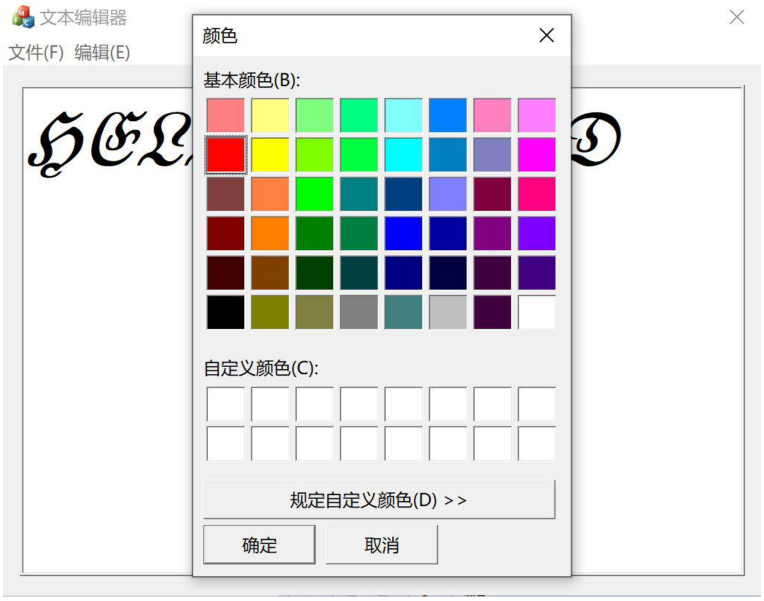
□ 下划线(U)

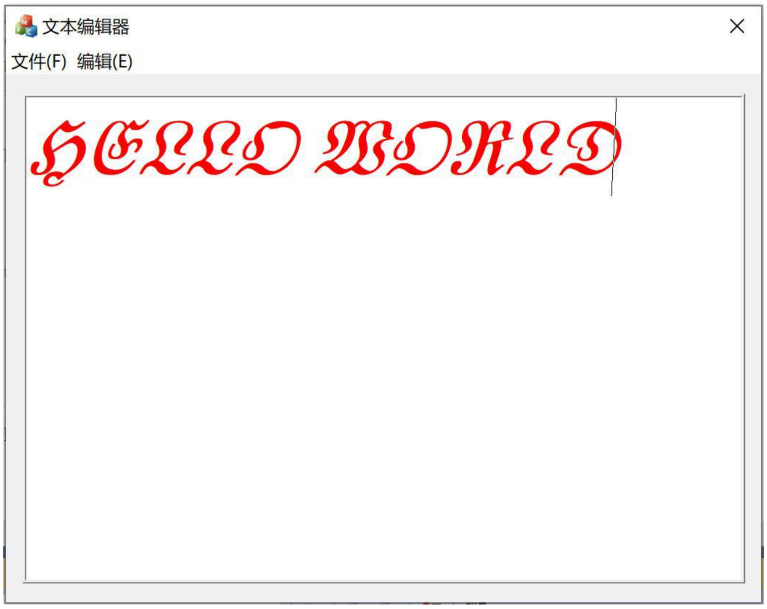
颜色(C): 黑色

示例

AaBb

字符集(R): 西欧语言





HELLO WORLD

2. 请观看视频“图形用户接口设计(演示)”，完整实现学生管理系统的所有功能。

```
class student
{
public:
	CString index;
	CString name;
	CString sex;
	CString birthday;
	CString hometown;
};
class Management
{
public:
	CString add[100];
	CString del[100];
	CString sort[100];

};

void CStudentDlg::OnBnClickedMale()
{
	m_sex = 0;
}

void CStudentDlg::OnBnClickedFemale()
{
	m_sex = 1;
}

void CStudentDlg::OnBnClickedAdd()
{
	UpdateData(true);
	CString sex;
	if (m_sex == 0)
		sex = "男";
	else if (m_sex == 1)
		sex = "女";
	m_list.InsertItem(i, m_index);
	m_list.SetItemText(i, 1, m_name);
	m_list.SetItemText(i, 2, m_sex);
	m_list.SetItemText(i, 3, m_birthday);
	m_list.SetItemText(i, 4, m_hometown);
	MessageBox(_T("添加成功！"));
```

```
    UpdateData(false);
}

void CStudentDlg::OnBnClickedChange()
{
    CListCtrl *plist = (CListCtrl *)GetDlgItem(IDC_LIST);
    POSITION pos = plist->GetFirstSelectedItemPosition();
    int nSel=plist->GetNextSelectedItem(pos);
    if (nSel < 0) {
        MessageBox(_T("请选中要修改的项！ "), MB_OK);
    }
    else {
        if (AfxMessageBox(_T("确认修改?"), MB_YESNO) == IDYES) {
            CString str;
            GetDlgItemText(IDC_INDEX, str);
            plist->SetItemText(nSel, 0, str);
            GetDlgItemText(IDC_NAME, str);
            plist->SetItemText(nSel, 1, str);
            if (m_sex == 0) {
                GetDlgItemText(IDC_MALE, str);
                plist->SetItemText(nSel, 2, str);
            }
            else if (m_sex == 1) {
               GetDlgItemText(IDC_FEMALE, str);
               plist->SetItemText(nSel, 2, str);
            }
            GetDlgItemText(IDC_BIRTH, str);
            plist->SetItemText(nSel, 3, str);
            GetDlgItemText(IDC_HOME, str);
            plist->SetItemText(nSel, 4, str);
        }
    }
}

void CStudentDlg::OnBnClickedDel()
{
    POSITION pos = m_list.GetFirstSelectedItemPosition();
    if (pos == NULL)
        return;
    else {
        while(pos) {
            int nItem = m_list.GetNextSelectedItem(pos);
            m_list.DeleteItem(nItem);
        }
```

```
	}
}

void CStudentDlg::SavetoFile(CString & strFilePath)
{
	USES_CONVERSION;
	CFile mytxtFile;
	CString strCaption,strMsg;
	if (!mytxtFile.Open(strFilePath, CFile::modeCreate | CFile::modeReadWrite)) {
		MessageBox(_T("打开失败"));
		return 0;
	}
	int width[5] = {0};
	int i=0, j=0, nLen=0;
	int nCount = m_list.GetItemCount();
	char format[512] = {0};
	char buf[1024] = {0};
	CString str0, str1, str2, str3, str4;
	str0.LoadString(IDC_INDEX);
	width[0] = strlen(T2A(str0));
	str1.LoadString(IDC_NAME);
	width[1]=strlen(T2A(str1));
	if (m_sex == 0) {
		str2.LoadString(IDC_MALE);
		width[2] = strlen(T2A(str2));
	}
	else if (m_sex == 1) {
		str2.LoadString(IDC_FEMALE);
		width[2] = strlen(T2A(str2));
	}
	str3.LoadString(IDC_BIRTH);
	width[3] = strlen(T2A(str3));
	str4.LoadString(IDC_HOME);
	width[4] = strlen(T2A(str4));
	sprintf_s(format, "%%-%ds %%-%ds %%-%ds %%-%ds %%-%ds %%-%ds \r\n"
			,width[0], width[1], width[2], width[3], width[4]);
	mytxtFile.Write(buf, strlen(buf));
	if (DoModal()==IDOK) {
		strPath=pFile->GetPathName();
	}
	CStdioFile file;
	CString strLine, temp;
	CStringArray strFile;
	int count = m_list.GetItemCount();
```

```
    setlocale(LC_CTYPE,("chs"));
    student stu;
    if (!file.Open(strPath,CFile::modeRead))
    {
        temp.Format(_T("%s 文件不存在!"), strPath);
        AfxMessageBox(temp);
    }
    while(file.ReadString(strLine))
    {
        AfxExtractSubString(stu.index, strLine, 0, '-');
        AfxExtractSubString(stu.name, strLine, 1, '-');
        AfxExtractSubString(stu.sex, strLine, 2, '-');
        AfxExtractSubString(stu.birthday, 3, '-');
        AfxExtractSubString(stu.hometown, strLine, 4, '-');
        m_list.InsertItem(count, stu.index);
        m_list.SetItemText(count, 1, stu.name);
        m_list.SetItemText(count, 2, stu.sex);
        m_list.SetItemText(count, 3, stu.birthday);
        m_list.SetItemText(count, 4, stu.hometown);
    }
    file.Close();
}
```

3. 有一个 MFC 应用程序，运行过程中可以随时通过鼠标画不同颜色大小的圆形。每个圆形以与水平成某角度做匀速直线运动，当遇到视图边界时反弹并继续运动。

```
// CBallView.h

#include <vector>
using namespace std;

const double walk = 5.0;
const double dirX[] = { walk, -walk, -walk, walk };
const double dirY[] = { -walk, -walk, walk, walk };

struct circle {
    double x, y, r, r1, g1, b1, r2, g2, b2;
    int dir;
    circle(double x, double y, double r) {
        this->x = x; this->y = y; this->r = r;
        this->r1 = rand() % 256;
        this->r2 = rand() % 256;
        this->g1 = rand() % 256;
        this->g2 = rand() % 256;
        this->b1 = rand() % 256;
        this->b2 = rand() % 256;
        dir = 0;
    }
};

extern vector<circle> circles;

class CBallView : public CView
{
......
public:
    afx_msg void OnLButtonDown(UINT nFlags, CPoint point);
    afx_msg void OnLButtonUp(UINT nFlags, CPoint point);
    virtual void OnInitialUpdate();
    afx_msg void OnTimer(UINT_PTR nIDEvent);
};
```

```
// CBallView.cpp
vector<circle> circles;

CBallView::CBallView() noexcept
{
    srand(time(0));
}

void CBallView::OnDraw(CDC* /*pDC*/)
{
    CDC* pDC = GetDC();
    pDC->SelectStockObject(NULL_BRUSH);
    pDC->SetROP2(R2_XORPEN);
    for (int i = 0; i < circles.size(); i++) {
        circle& c = circles[i];
        CBrush fillbrush;
        pDC->SelectObject(CPen(0, 1, RGB(c.r2, c.g2, c.b2)));
        pDC->Ellipse(c.x - c.r, c.y - c.r, c.x + c.r, c.y + c.r);
        fillbrush.CreateSolidBrush(RGB(c.r1, c.g1, c.b1));
        pDC->SelectObject(fillbrush);
        const int len = 5;
        pDC->Ellipse(c.x - c.r + len, c.y - c.r + len, c.x + c.r - len, c.y + c.r - len);
    }
    ReleaseDC(pDC);
}
```

```
void CBallView::OnLButtonDown(UINT nFlags, CPoint point)
{
    flag_LBTNDown = true;
    m_pStart = point;
    CView::OnLButtonDown(nFlags, point);
}

void CBallView::OnLButtonUp(UINT nFlags, CPoint point)
{
    if (flag_LBTNDown) {
        CPoint center;
        double nRadius;
        center.x = (float(m_pStart.x + point.x)) / 2;
        center.y = (float(m_pStart.y + point.y)) / 2;
        nRadius = sqrt((double)(point.y - m_pStart.y) * (point.y - m_pStart.y) +
            (point.x - m_pStart.x) * (point.x - m_pStart.x)) / 2;
        circles.push_back(circle(center.x, center.y, nRadius));
        flag_LBTNDown = false;
    }
    CView::OnMouseMove(nFlags, point);
}

void CBallView::OnInitialUpdate()
{
    CView::OnInitialUpdate();
    SetTimer(1, 10, NULL);
}


void CBallView::OnTimer(UINT_PTR nIDEvent)
{
    if (nIDEvent == 1) {
        CRect rcClient;
        GetClientRect(rcClient);
        for (int i = 0; i < circles.size(); i++) {
            circle& c = circles[i];
            c.x += dirX[c.dir]; c.y += dirY[c.dir];
            double left = c.x - c.r, right = c.x + c.r;
            double up = c.y - c.r, down = c.y + c.r;
            if (left <= 0) {
                if (c.dir == 1) { c.dir = 0; continue; }
                if (c.dir == 2) { c.dir = 3; continue; }
            }
            if (right >= rcClient.Width()) {
                if (c.dir == 0) { c.dir = 1; continue; }
                if (c.dir == 3) { c.dir = 2; continue; }
            }
            if (up <= 0) {
                if (c.dir == 0) { c.dir = 3; continue; }
                if (c.dir == 1) { c.dir = 2; continue; }
            }
            if (down >= rcClient.Height()) {
                if (c.dir == 2) { c.dir = 1; continue; }
                if (c.dir == 3) { c.dir = 0; continue; }
            }
        }
        Invalidate();
    }
    CView::OnTimer(nIDEvent);
}
```