**郑州轻工业大学**

**课程报告**

**《C#程序设计》**

学生姓名： 原彬贺

学生学号： 542001020223

专业班级： 计算机20-02班

任课老师： 吴庆岗

评定成绩：

目录

[概述 3](#_Toc89466138)

[课程报告主要内容 3](#_Toc89466139)

[开发工具选择 3](#_Toc89466140)

[开发环境 3](#_Toc89466141)

[需求分析 3](#_Toc89466142)

[系统需求简介 3](#_Toc89466143)

[系统目标 3](#_Toc89466144)

[功能需求分析 3](#_Toc89466145)

[非功能性需求分析 3](#_Toc89466146)

[总体设计 3](#_Toc89466147)

[设计概述 3](#_Toc89466148)

[系统总体结构及功能模块划分 3](#_Toc89466149)

[院系信息管理模块 3](#_Toc89466150)

[学生信息管理模块 3](#_Toc89466151)

[详细设计 3](#_Toc89466152)

[概述 4](#_Toc89466153)

[系统程序流程图 4](#_Toc89466154)

[系统主要功能模块简介 4](#_Toc89466155)

[主要功能模块代码 4](#_Toc89466156)

[登录界面代码设计 4](#_Toc89466157)

[查询模块代码设计 4](#_Toc89466158)

[课程学习收获 4](#_Toc89466159)

# 概述

## 课程报告主要内容

设计一个系统，使其具有较为完善的效果

## 开发工具选择

Vs2019

## 开发环境

# 需求分析

## 系统需求简介

### 系统目标

设计一个适合编写代码的文本编辑软件，使其具有一些市场上常见文本编辑软件不具有的功能。包含长文本中词组替换，含数字文本数字自递增，长文本短语寻查，多文本复制，文本文件操作。

### 功能需求分析

1. 长文本中词组替换

在日常编写代码时，由于每个功能可能仅是其中代码的名称的位置不同，对于长文本替换单词来说，如果需要替换多个且含有把a替换到b，把b替换到a时，其余系统只可以手动先把a替换到c再把b替换到a，最后把c替换到b。如果涉及到大量替换，不仅枯燥麻烦，还容易由于粗心而犯错。

1. 含数字文本数字自递增

在日常开发时，由于代码的重复性，有时候仅需要讲代码里的数字加一便是下一个按键的内容，比如多个按键对应对个文本框，代码均类似，进名称不同，且名称中的数字仅增加一，耗时且麻烦。而本功能，只需要将需更改的内容复制到剪贴板，然后点击一下递增按键，剪贴板中的数字内容便自动增加，无需进行多余操作

1. 长文本短语寻查

日常使用中，编写代码后，想查看某一块功能的代码，若是代码量大，则不易查找，耗费时间，但是使用本功能，便可快速寻找到该功能代码。本系统有两种操作方式，一是通过按键直接打开文本内容便可将文本内容复制到文本框中，而是将文本内容直接复制到文本框。点击搜寻，输入搜寻内容，点确定，搜寻内容字体变变为红色，在文本框中更改内容后，可将其保存到电脑，防止数据丢失

1. 多文本复制

在日常使用中，由于多次用到多个内容，若将其复制到笔记软件，则需要进行比较复杂的操作，而本功能，只需将文本复制一下，点击粘贴，文本便复制到文本框里，点击复制其对应文本框中的内容便复制到剪贴板中。若有多个文本，可将其粘贴到剩余多个文本框再进行操作。

1. 临时文本文件操作

在日常使用中，有时仅是临时使用一段文本，但是一本文件系统会将其保存，耗费空间且日后不易删除。本系统便可将其复制到文本框，若需要进行保存，点保存即可，若用完后想删除，直接点删除，该文件便删除，防止有时创建文件后找不到并懒得删的问题

# 总体设计

## 设计概述

设计两个模块，分别是文本和文件模块，再在文本和文件模块中添加其功能。

## 系统总体结构及功能模块划分

### 文件系统模块

长文本短语寻查

临时文本文件操作

### 文本系统模块

长文本中词组替换

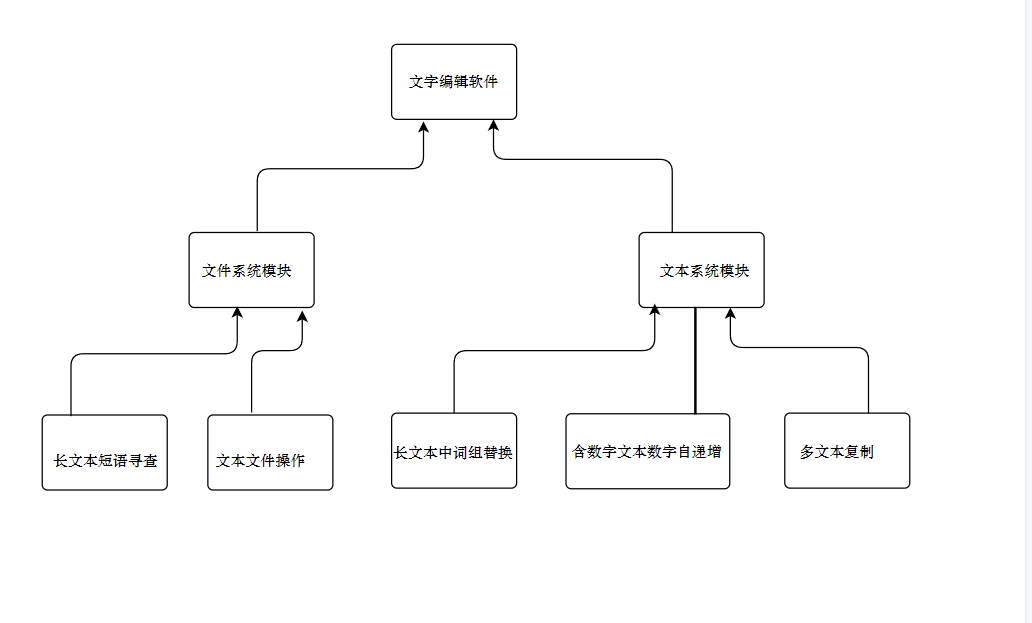
含数字文本数字自递增

多文本复制

# 详细设计

## 概述

## 系统程序流程图



## 系统主要功能模块简介

1. 主界面，如图1所示

主界面含菜单和进入功能窗口按键。其中菜单含有文件文本和设置。文件中包含长文本短语寻查，临时文本文件操作，文本包含长文本中词组替换、含数字文本数字自递增、多文本复制，设置中含退出键。



1. 长文本中词组替换

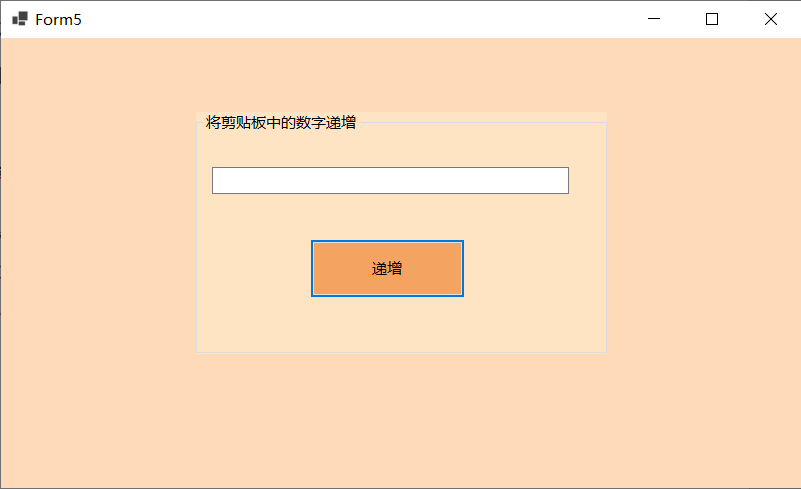


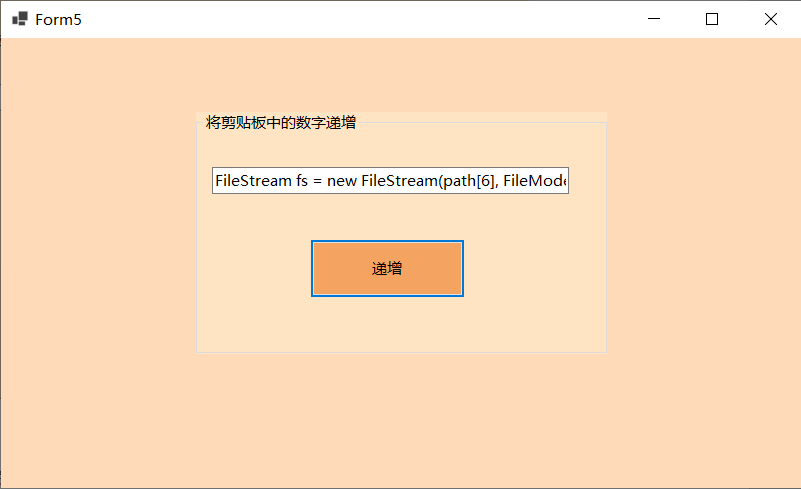


点击文本粘贴，剪贴板中的文本便粘贴到左侧文本框。

再点击单词输入，在其上写入原单词词组，右侧写入替换后的词组，若需要再次替换，点击再次输入，若输入完毕，点击输入完成，左侧文本框中的内容即为替换后的文本，点击复制文本，即复制到剪贴板

1. 含数字文本数字自递增





例如剪贴板上的内容为FileStream fs = new FileStream(path[5], FileMode.Create, FileAccess.Write);

string stext = textBox6.Text;

byte[] btext = Encoding.Default.GetBytes(stext);

fs.Write(btext, btext.Length);

fs.Close();，直接点击递增按键，剪贴板中和文本框中的内容便为FileStream fs = new FileStream(path[6], FileMode.Create, FileAccess.Write);

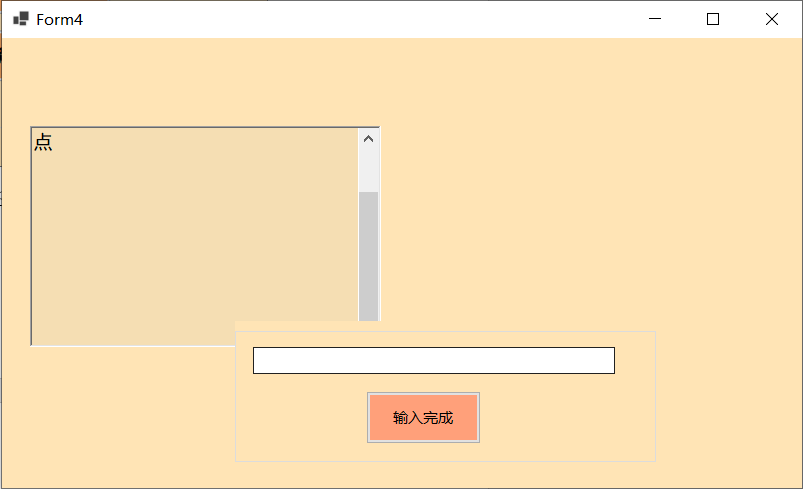
string stext = textBox7.Text;

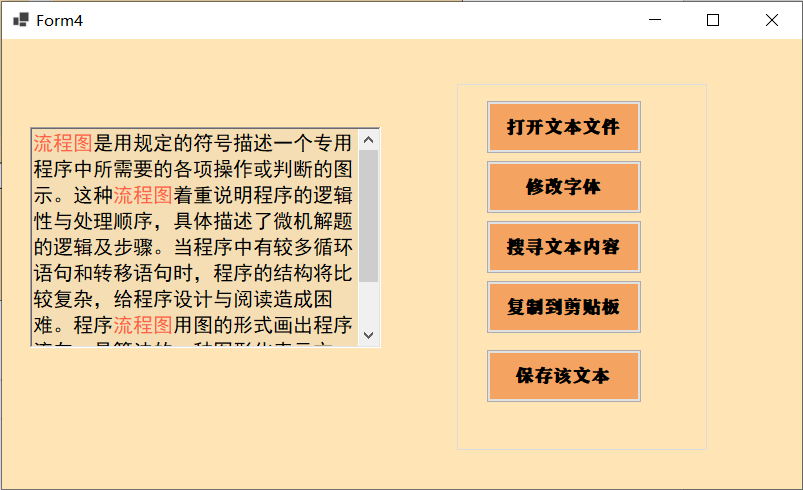
byte[] btext = Encoding.Default.GetBytes(stext);

fs.Write(btext, btext.Length); fs.Close();，不需要进行先将剪贴板中的内容复制到文本框，点击递增后再进行复制，该功能极为简洁实用。

1. 长文本短语寻找功能







可打开文本，文本内容便复制到左侧的文本区域，也可自行进行复制到文本框，

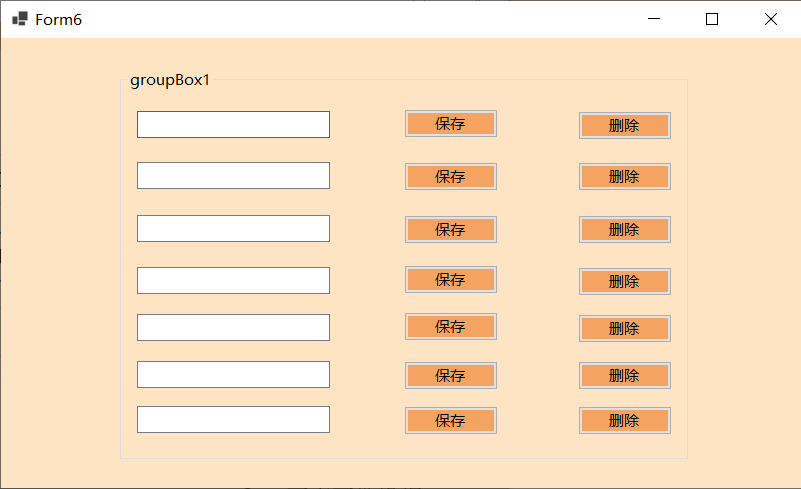
选中文本后可修改字体，点击搜寻文本内容，输入搜寻内容，点击输入完成，该关键字便变为红色。可对其进行复制，也可将其保存到文件。

1. 多文本复制



点击粘贴便将剪贴板中的内容粘贴到其对应的文本框，点击复制可将该文本复制到剪贴板，若有多个文本，且需要间隔操作，该功能便非常方便。

1. 文本文件操作



将文本复制到文本框后，便可对其进行保存，无需进行选择路径，直接点击保存即可，用完后点击删除便可

# 主要功能模块代码

## 主界面代码设计

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace 实验

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void timer1\_Tick(object sender, EventArgs e)

{

}

private void 类别ToolStripMenuItem\_Click(object sender, EventArgs e)

{

}

private void 单词替换ToolStripMenuItem\_Click(object sender, EventArgs e)

{

Form2 form2 = new Form2();

form2.Show();

}

private void 剪贴短文本ToolStripMenuItem\_Click(object sender, EventArgs e)

{

Form3 form3 = new Form3();

form3.Show();

}

private void 长时文本ToolStripMenuItem\_Click(object sender, EventArgs e)

{

}

private void 临时文本ToolStripMenuItem\_Click(object sender, EventArgs e)

{

Form6 form6 = new Form6();

form6.Show();

}

private void 单词插入ToolStripMenuItem\_Click(object sender, EventArgs e)

{

Form5 form5 = new Form5();

form5.Show();

}

private void Form1\_Load(object sender, EventArgs e)

{

}

private void 退出ToolStripMenuItem\_Click(object sender, EventArgs e)

{

this.Close();

}

private void menuStrip1\_ItemClicked(object sender, ToolStripItemClickedEventArgs e)

{

}

private void toolStripContainer1\_ContentPanel\_Load(object sender, EventArgs e)

{

}

private void button1\_Click(object sender, EventArgs e)

{

Form3 form3 = new Form3();

form3.Show();

}

private void button3\_Click(object sender, EventArgs e)

{

Form6 form6 = new Form6();

form6.Show();

}

private void button2\_Click(object sender, EventArgs e)

{

Form2 form2 = new Form2();

form2.Show();

}

private void button4\_Click(object sender, EventArgs e)

{

Form5 form5 = new Form5();

form5.Show();

}

private void button5\_Click(object sender, EventArgs e)

{

this.Close();

}

private void 退出ToolStripMenuItem\_Click\_1(object sender, EventArgs e)

{

this.Close();

}

private void 文本替换ToolStripMenuItem\_Click(object sender, EventArgs e)

{

Form2 form2 = new Form2();

form2.Show();

}

private void 文本插入ToolStripMenuItem\_Click(object sender, EventArgs e)

{

Form5 form5 = new Form5();

form5.Show();

}

private void 临时文本ToolStripMenuItem\_Click\_1(object sender, EventArgs e)

{

Form6 form6 = new Form6();

form6.Show();

}

private void 多文本剪贴ToolStripMenuItem\_Click(object sender, EventArgs e)

{

Form3 form3 = new Form3();

form3.Show();

}

private void 文件读取ToolStripMenuItem\_Click(object sender, EventArgs e)

{

Form4 form4 = new Form4();

form4.Show();

}

private void button6\_Click(object sender, EventArgs e)

{

Form4 form4 = new Form4();

form4.Show();

}

private void Form1\_KeyDown(object sender, KeyEventArgs e)

{

}

}

}

## 长文本中词组替换

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace 实验

{

public partial class Form2 : Form

{

public Form2()

{

InitializeComponent();

}

public string date;

private void button1\_Click(object sender, EventArgs e)

{

groupBox1.Visible = false;

groupBox2.Visible = true;

date = richTextBox1.Text;

}

public List<string> dateben = new List<string>();

public List<string> datenew = new List<string>();

private void label1\_Click(object sender, EventArgs e)

{

}

private void textBox1\_TextChanged(object sender, EventArgs e)

{

}

private void label3\_Click(object sender, EventArgs e)

{

}

private void textBox3\_TextChanged(object sender, EventArgs e)

{

}

private void button2\_Click(object sender, EventArgs e)

{

dateben.Add(textBox1.Text);

datenew.Add(textBox2.Text);

textBox1.Text = "";

textBox2.Text = "";

}

private void button3\_Click(object sender, EventArgs e)

{

dateben.Add(textBox1.Text);

datenew.Add(textBox2.Text);

groupBox1.Visible = true;

groupBox2.Visible = false;

string mid;

for (int i = 0; i < dateben.Count; i++)

{

mid = Convert.ToString(i);

date=date.Replace(dateben[i], "aaa" + mid);

}

for (int i = 0; i < datenew.Count; i++)

{

mid = Convert.ToString(i);

date=date.Replace("aaa" + mid,datenew[i] );

}

richTextBox1.Text = date;

}

private void label2\_Click(object sender, EventArgs e)

{

}

private void Form2\_Load(object sender, EventArgs e)

{

}

private void groupBox2\_Enter(object sender, EventArgs e)

{

}

private void button4\_Click(object sender, EventArgs e)

{

Clipboard.SetDataObject(richTextBox1.Text);

}

private void button5\_Click(object sender, EventArgs e)

{

richTextBox1.Text = Clipboard.GetText();

}

}

}

## 含数字文本数字自递增

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace 实验

{

public partial class Form5 : Form

{

public Form5()

{

InitializeComponent();

}

private void button1\_Click(object sender, EventArgs e)

{

string dizeng;

dizeng = Clipboard.GetText();

char[] dizengt= dizeng.ToCharArray();

int a,len=dizeng.Length;

int j = 0;

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

{

string ceshi = Convert.ToString(dizengt[i]);

if (int.TryParse(ceshi, out a))

{

dizeng = dizeng.Remove(j, 1);

if (a == 9)

{

int c;

if (i >= 1)

{

if (int.TryParse(Convert.ToString(dizengt[i - 1]), out c))

{

if (c == 9)

{

dizeng = dizeng.Remove(j - 1, 1);

dizeng = dizeng.Insert(j - 1, "100");

j++;

}

else

{

dizeng = dizeng.Remove(j - 1, 1);

dizeng = dizeng.Insert(j - 1, Convert.ToString((c + 1) \* 10));

}

}

else

{

dizeng = dizeng.Insert(j, "10");

j++;

}

}

}

else

{

if (i >= 1)

{

int c;

if (int.TryParse(Convert.ToString(dizengt[i - 1]), out c))

{

dizeng = dizeng.Remove(j-1, 1);

dizeng = dizeng.Insert(j - 1, Convert.ToString(dizengt[i - 1]));

dizeng = dizeng.Insert(j, Convert.ToString(a + 1));

}

else

dizeng = dizeng.Insert(j, Convert.ToString(a + 1));

}

}

}

j++;

}

//dizeng = dizeng.Insert(2, "34");

textBox1.Text = dizeng;

Clipboard.SetDataObject(textBox1.Text);

// textBox1.Text = Convert.ToString(dizengt[0]);

}

private void textBox1\_TextChanged(object sender, EventArgs e)

{

}

private void Form5\_Load(object sender, EventArgs e)

{

}

private void button2\_Click(object sender, EventArgs e)

{

Clipboard.SetDataObject(textBox1.Text);

}

private void groupBox1\_Enter(object sender, EventArgs e)

{

}

}

}

## 长文本短语寻找功能

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.IO;

namespace 实验

{

public partial class Form4 : Form

{

public Form4()

{

InitializeComponent();

}

private void treeView1\_AfterSelect(object sender, TreeViewEventArgs e)

{

}

public string seek;

private void button1\_Click(object sender, EventArgs e)

{

DialogResult dr = openFileDialog1.ShowDialog();

if (dr == DialogResult.OK)

{

string path = openFileDialog1.FileName;

FileStream fd = new FileStream(path, FileMode.Open, FileAccess.Read);

byte[] mo = new byte[fd.Length];

fd.Read(mo, 0, mo.Length);

richTextBox1.Text = Encoding.Default.GetString(mo);

fd.Close();

}

}

private void button4\_Click(object sender, EventArgs e)

{

DialogResult sr = saveFileDialog1.ShowDialog();

if (sr == DialogResult.OK)

{

string na = saveFileDialog1.FileName;

FileStream sa = new FileStream(na, FileMode.Create);

string stext = richTextBox1.Text;

byte[] btext = Encoding.Default.GetBytes(stext);

sa.Write(btext, 0, btext.Length);

sa.Close();

}

}

private void button2\_Click(object sender, EventArgs e)

{

groupBox1.Visible = false;

groupBox2.Visible = true;

}

private void toolStripLabel1\_Click(object sender, EventArgs e)

{

}

private void button5\_Click(object sender, EventArgs e)

{

seek = textBox1.Text;

groupBox2.Visible = false;

groupBox1.Visible = true;

int[] a;

int j,i = 0;

string text = richTextBox1.Text;

j = text.Length;

while (text.IndexOf(seek,i,j)!=-1)

{

i = text.IndexOf(seek,i,j);

richTextBox1.Select(i, seek.Length);

richTextBox1.SelectionColor = Color.Tomato;

i = i + seek.Length;

j = text.Length - i;

if (i + seek.Length == text.Length)

break;

}

}

private void button3\_Click(object sender, EventArgs e)

{

Clipboard.SetDataObject(richTextBox1.Text);

}

private void button6\_Click(object sender, EventArgs e)

{

fontDialog1.ShowDialog();

richTextBox1.SelectionFont = fontDialog1.Font;

}

private void groupBox2\_Enter(object sender, EventArgs e)

{

}

}

}

## 多文本复制

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace 实验

{

public partial class Form3 : Form

{

public Form3()

{

InitializeComponent();

}

private void treeView1\_AfterSelect(object sender, TreeViewEventArgs e)

{

}

private void button1\_Click(object sender, EventArgs e)

{

if (textBox1.Text.Length != 0)

{ Clipboard.SetDataObject(textBox1.Text); }

}

private void button2\_Click(object sender, EventArgs e)

{

if (textBox2.Text.Length != 0)

{ Clipboard.SetDataObject(textBox2.Text); }

}

private void button3\_Click(object sender, EventArgs e)

{

if (textBox3.Text.Length != 0)

{ Clipboard.SetDataObject(textBox3.Text); }

}

private void button4\_Click(object sender, EventArgs e)

{

if (textBox4.Text.Length != 0)

{ Clipboard.SetDataObject(textBox4.Text); }

}

private void button5\_Click(object sender, EventArgs e)

{

if (textBox5.Text.Length != 0)

{ Clipboard.SetDataObject(textBox5.Text); }

}

private void button6\_Click(object sender, EventArgs e)

{

if (textBox6.Text.Length != 0)

{ Clipboard.SetDataObject(textBox6.Text); }

}

private void Form3\_Load(object sender, EventArgs e)

{

}

private void label1\_Click(object sender, EventArgs e)

{

}

private void button7\_Click(object sender, EventArgs e)

{

textBox1.Text = Clipboard.GetText();

}

private void button8\_Click(object sender, EventArgs e)

{

textBox2.Text = Clipboard.GetText();

}

private void button9\_Click(object sender, EventArgs e)

{

textBox3.Text = Clipboard.GetText();

}

private void button10\_Click(object sender, EventArgs e)

{

textBox4.Text = Clipboard.GetText();

}

private void button11\_Click(object sender, EventArgs e)

{

textBox5.Text = Clipboard.GetText();

}

private void button12\_Click(object sender, EventArgs e)

{

textBox6.Text = Clipboard.GetText();

}

}

}

## 文本文件操作

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.IO;

namespace 实验

{

public partial class Form6 : Form

{

public Form6()

{

InitializeComponent();

}

private void Form6\_Load(object sender, EventArgs e)

{

}

public string[] path = new string[] { @"E:/文本/texi11.txt",@"E:/文本/texi12.txt" , @"E:/文本/texi13.txt" , @"E:/文本/texi13.txt" , @"E:/文本/texi14.txt" , @"E:/文本/texi15.txt" , @"E:/文本/texi16.txt" };

private void button1\_Click(object sender, EventArgs e)

{

FileStream fs = new FileStream(path[0], FileMode.Create, FileAccess.Write);

string stext = textBox1.Text;

byte[] btext = Encoding.Default.GetBytes(stext);

fs.Write(btext, 0, btext.Length);

fs.Close();

}

private void button2\_Click(object sender, EventArgs e)

{

File.Delete(path[0]);

}

private void textBox1\_TextChanged(object sender, EventArgs e)

{

}

private void button3\_Click(object sender, EventArgs e)

{

FileStream fs = new FileStream(path[1], FileMode.Create, FileAccess.Write);

string stext = textBox2.Text;

byte[] btext = Encoding.Default.GetBytes(stext);

fs.Write(btext,0 , btext.Length);

fs.Close();

}

private void button4\_Click(object sender, EventArgs e)

{

FileStream fs = new FileStream(path[2], FileMode.Create, FileAccess.Write);

string stext = textBox3.Text;

byte[] btext = Encoding.Default.GetBytes(stext);

fs.Write(btext, 0, btext.Length);

fs.Close();

}

private void button5\_Click(object sender, EventArgs e)

{

FileStream fs = new FileStream(path[3], FileMode.Create, FileAccess.Write);

string stext = textBox4.Text;

byte[] btext = Encoding.Default.GetBytes(stext);

fs.Write(btext,0 , btext.Length);

fs.Close();

}

private void button6\_Click(object sender, EventArgs e)

{

FileStream fs = new FileStream(path[4], FileMode.Create, FileAccess.Write);

string stext = textBox5.Text;

byte[] btext = Encoding.Default.GetBytes(stext);

fs.Write(btext,0 , btext.Length);

fs.Close();

}

private void button7\_Click(object sender, EventArgs e)

{

FileStream fs = new FileStream(path[5], FileMode.Create, FileAccess.Write);

string stext = textBox6.Text;

byte[] btext = Encoding.Default.GetBytes(stext);

fs.Write(btext,0 , btext.Length);

fs.Close();

}

private void button8\_Click(object sender, EventArgs e)

{

FileStream fs = new FileStream(path[6], FileMode.Create, FileAccess.Write);

string stext = textBox7.Text;

byte[] btext = Encoding.Default.GetBytes(stext);

fs.Write(btext, 0, btext.Length);

fs.Close();

}

private void button9\_Click(object sender, EventArgs e)

{

File.Delete(path[1]);

}

private void button10\_Click(object sender, EventArgs e)

{

File.Delete(path[2]);

}

private void button11\_Click(object sender, EventArgs e)

{

File.Delete(path[3]);

}

private void button12\_Click(object sender, EventArgs e)

{

File.Delete(path[4]);

}

private void button13\_Click(object sender, EventArgs e)

{

File.Delete(path[5]);

}

}

}

# 课程学习收获

通过本次大作业，在编写过程中，学会了将一个个form窗口联系起来，由于需要将多个窗口同时进行同时操作，则将其设置为非模式窗口展示。在具体的文件编写时，由于对于一些想实现的功能所需的代码并不熟悉比如将文本框中的内容复制到剪贴板，通过自行查找了相关代码，成功实现了文本从剪贴板到文本框再从文本框到剪贴板的数据传输。在对文本进行替换时，通过将文本内容转化为字符串数组对其替换寻查。这次大作业我最主要的收获就是碰到想实现的功能却不知道相关代码的问题时，不要因为知识的匮乏而降低自己对想法的要求，应该补充自己的知识去实现自己的想法。为之则易，不为则难！

|  |
| --- |
| **评语**（不要删除，指导老师填写评语，放置在实验报告后面单独成页） |
|  |