**《C#程序设计》课程实验报告**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **学号：** | **12021051064** |  | **姓名：** | **华心童** |

**实验十五、文件（2）**

**一、实验目的**

1、了解文件流的操作。

2、将实验报告云盘，文件名是**学号姓名实验15**命名。

**二、实验内容**

实验准备：在F盘创建学号文件夹，用于存放实验结果文件。

运行VS2013，创建解决方案是“**Ex学号-15**”。

1. **基本实验**

**任务1、文件读写【项目名P1501】**

**要求：**使用StreamWriter写入一些逻辑型、数值型、字符串或长文本，并通过StreamReader利用多种方式（单个字符、一行等）读出内容。

**程序源程序代码：**粘贴program.cs中代码

string path = @"D:\jason1.txt";

FileStream fs1 = new FileStream(path,FileMode.OpenOrCreate);

StreamWriter sw1 = new StreamWriter(fs1);

sw1.WriteLine(true);

sw1.WriteLine(88);

sw1.WriteLine("Apple");

sw1.Close();

fs1.Close();

//读入一行

FileStream fs2 = new FileStream(path,FileMode.Open);

StreamReader sr1 = new StreamReader(fs2);

string line = sr1.ReadLine();

while(line!=null)

{

Console.WriteLine("一行读出"+line);

line = sr1.ReadLine();

}

sr1.Close();

fs2.Close();

//读入一个字节

FileStream fs3 = new FileStream(path, FileMode.Open);

StreamReader sr2 = new StreamReader(fs3);

int content = sr2.Read();

while (content != -1)

{

Console.WriteLine("单个字符读出"+Convert.ToChar(content));

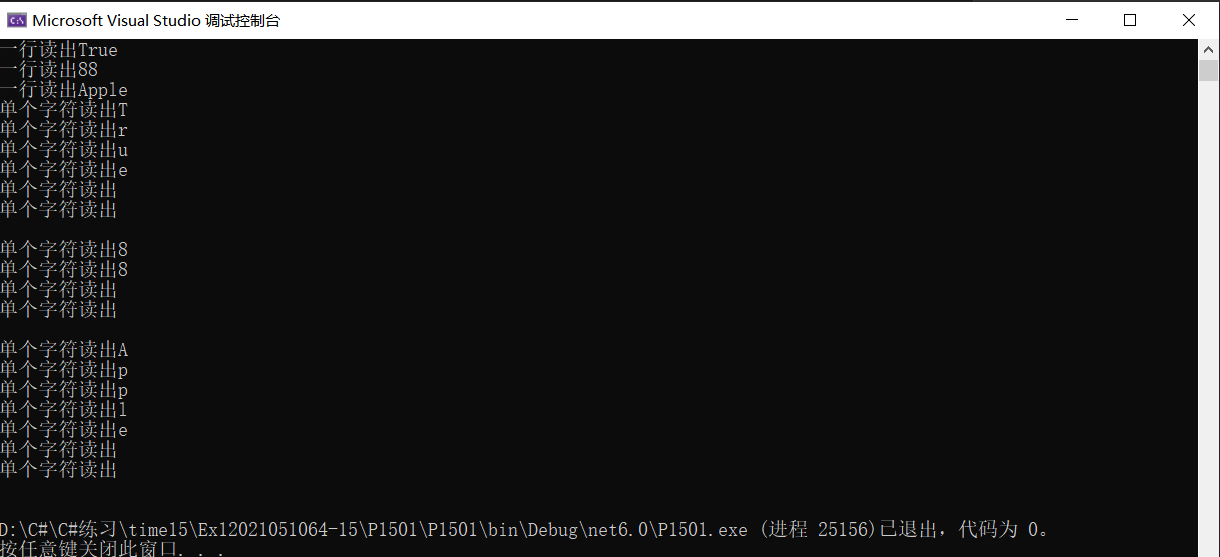
content = sr2.Read();

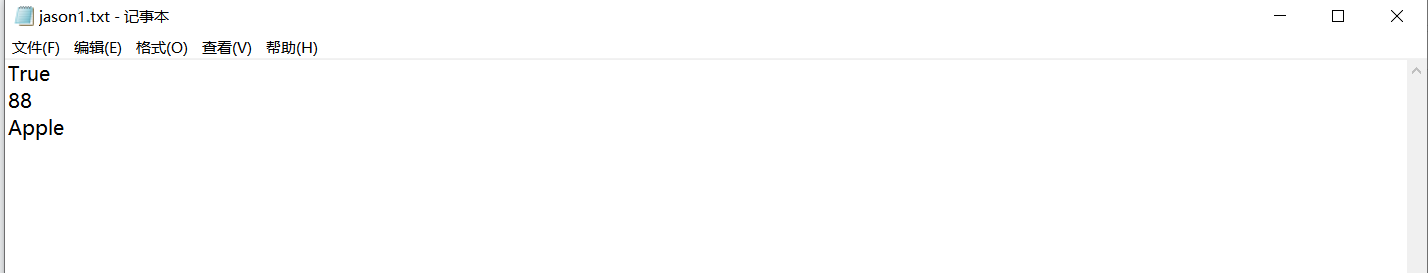
}

sr2.Close();

fs3.Close();

//粘贴运行结果界面截图





**任务2、文件追加，并统计某个字符的个数。【项目名P1502】**

**要求**：追加一段字符，完成字符的统计。

**程序源程序代码：**粘贴program.cs中代码

int count = 0;

StreamWriter sw2 = new StreamWriter(path,true);

sw2.WriteLine("你还是你吗");

sw2.Close();

FileStream fs4 = new FileStream(path, FileMode.Open);

StreamReader sr3 = new StreamReader(fs4);

content = sr3.Read();

while(content!=-1)

{

if (Convert.ToChar(content) == '你') count++;

content = sr3.Read();

}

sr3.Close();

fs4.Close();

Console.WriteLine("我共出现了{0}次",count);

//粘贴运行结果界面截图



**任务3、二进制流【P1503】**

要求：通过BinaryWriter写入文件并通过BinaryReader输出。

**程序源程序代码：**粘贴program.cs中代码

string path = @"D:\durant1.txt";

string target = @"D:\curry1.txt";

FileStream fs1 = new FileStream(path,FileMode.Create);

BinaryWriter bw1 = new BinaryWriter(fs1);

bw1.Write(1000);

bw1.Write(DateTime.Now.ToString());

bw1.Write("你好，同学");

bw1.Flush();

bw1.Close();

fs1.Close();

FileStream fs2 = new FileStream(path,FileMode.Open);

FileStream fs3 = new FileStream(target, FileMode.OpenOrCreate);

BinaryReader br1 = new BinaryReader(fs2);

BinaryWriter bw2 = new BinaryWriter(fs3);

for (int i = 0; i < br1.BaseStream.Length; i++)

{

bw2.Write(br1.ReadByte());

}

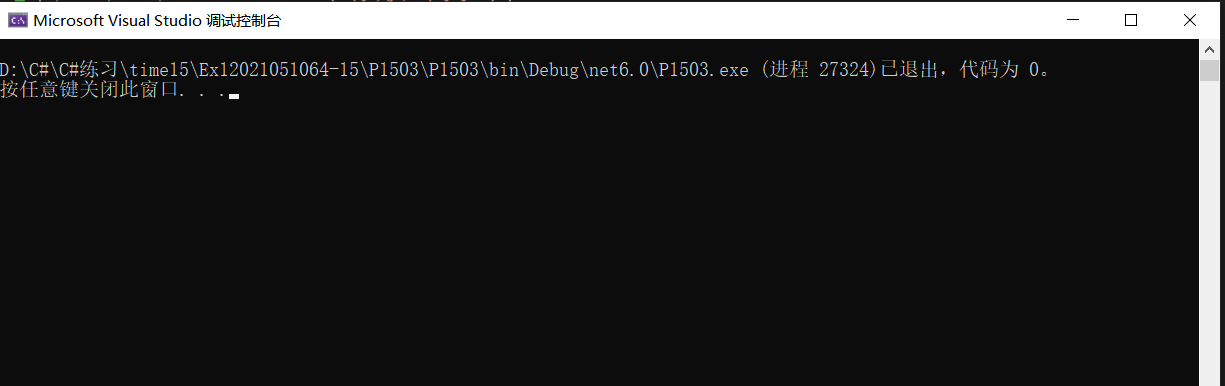
bw1.Close();

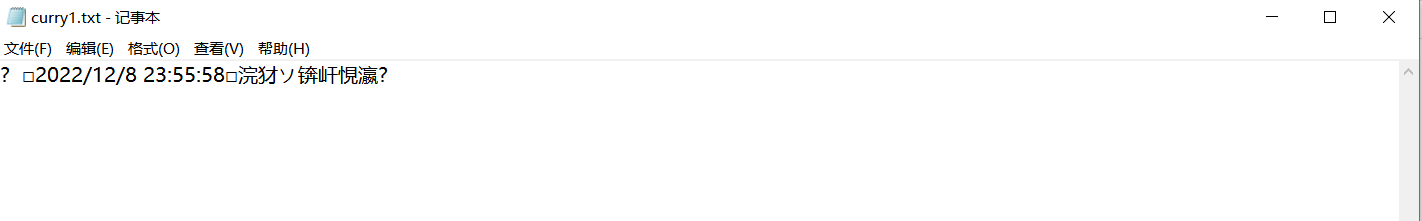
bw2.Close();

fs2.Close();

fs3.Close();

//粘贴运行结果界面截图





**任务4、文件读写【P1504】**

**要求**：利用二进制流，通过double型数组，写入一段成绩进入score.dat，将score.dat复制到scorebak.dat，并通过二进制流读出第2门和第5门成绩。

**程序源程序代码：**粘贴program.cs中代码

string path1 = @"D:\score.dat";

string path2 = @"D:\scorebak.dat";

double[] score = { 85.5, 78.6, 59.7, 89.2, 45.2 };

FileStream fs1 = new FileStream(path1,FileMode.OpenOrCreate);

BinaryWriter bw1 = new BinaryWriter(fs1);

for (int i = 0; i < 5; i++)

{

bw1.Write(score[i]);

}

bw1.Flush();

bw1.Close();

fs1.Close();

FileStream fs2 = new FileStream(path1, FileMode.Open);

FileStream fs3 = new FileStream(path2, FileMode.OpenOrCreate);

BinaryReader br1 = new BinaryReader(fs2);

BinaryWriter bw2 = new BinaryWriter(fs3);

for (int i = 0; i < br1.BaseStream.Length; i++)

{

bw2.Write(br1.ReadByte());

}

bw1.Close();

bw2.Close();

fs3.Close();

FileStream fs4 = new FileStream(path2, FileMode.OpenOrCreate,FileAccess.Read);

BinaryReader br2 = new BinaryReader(fs4);

Console.WriteLine("-------分界线-------");

Console.Write("输入第2门成绩:");

br1.BaseStream.Seek(8,SeekOrigin.Begin);

Console.WriteLine(br1.ReadDouble());

Console.WriteLine("-------分界线-------");

Console.Write("输入第5门成绩:");

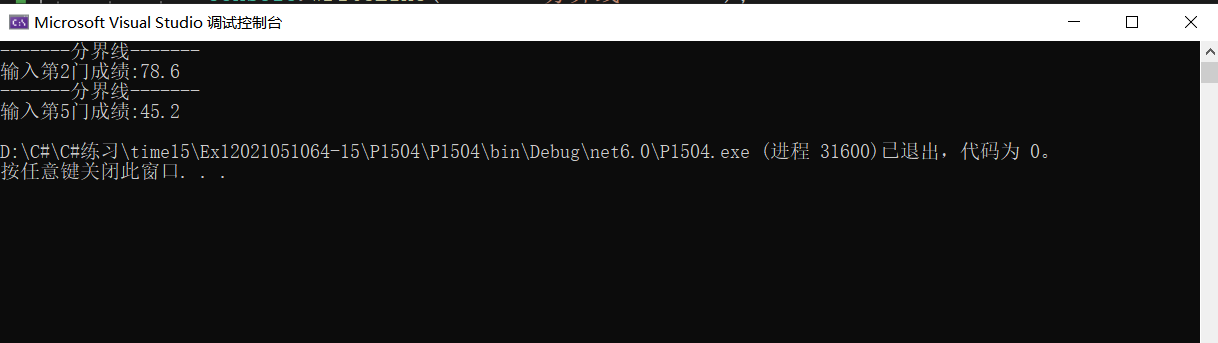
br1.BaseStream.Seek(32, SeekOrigin.Begin);

Console.WriteLine(br1.ReadDouble());

br2.Close();

fs4.Close();

//粘贴运行结果界面截图



**扩展任务5、序列化与反序列化【P1505】**

**要求**：创建Person类，通过序列化和反序列化读出实例化的Person类的信息。

**程序源程序代码：**粘贴program.cs中代码

[Serializable]

class Person

{

private string name;

public string Name

{

get { return name; }

set { name = value; }

}

private int age;

public int Age

{

get { return age; }

set { age = value; }

}

public override string ToString()

{

return string.Format("姓名：{0},年龄：{1}",this.Name,this.Age);

}

}

class Program

{

static void Main(string[] args)

{

Person person= new Person();

person.Name = "华心童";

person.Age = 20;

BinaryFormatter bf= new BinaryFormatter();

using(FileStream fs=new FileStream("D:\\Data.txt", FileMode.OpenOrCreate, FileAccess.ReadWrite))

{

bf.Serialize(fs, person);

Console.WriteLine("序列化操作成功，对象已写入文件");

}

using(FileStream fs1=new FileStream("D:\\Data.txt",FileMode.OpenOrCreate,FileAccess.ReadWrite))

{

object obj= bf.Deserialize(fs1);

Console.WriteLine("反序列化对象数据为"+obj);

}

person.ToString();

Console.ReadKey();

}

}

//粘贴运行结果界面截图

