Министерство образования Республики Беларусь

Учреждение образования

«Брестский государственный технический университет»

Кафедра ИИТ

ОТЧЕТ

по лабораторной работе №2

по дисциплине СПП

Выполнил:

студ. гр.ПО-3

Аниськин К.Д.

Проверил:

Крощенко А.А.

Брест 2020

Цель работы: приобрести базовые навыки работы с файловой системой в C#.

**Задание 1. Напишите программу, которая выводит слова, располагая их в порядке убывания частоты их появления. Перед каждым словом должно быть число его появлений.**

**Реализация алгоритмов:**

static void Main(string[] args)

{

string directory = Directory.GetCurrentDirectory() + "/";

string fileName = "file.txt";

string[] words = new string[0];

int[] wordsCount = new int[0];

int size = 0;

bool check;

using (StreamReader fs = new StreamReader($"{directory}{fileName}"))

{

while (!fs.EndOfStream)

{

check = false;

string[] temp = fs.ReadLine().Split(' ');

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

{

for (int j = 0; j < words.Length; j++)

{

if (temp[i] == words[j])

{

check = true;

wordsCount[j]++;

break;

}

}

if (!check)

{

size++;

Array.Resize(ref words, size);

Array.Resize(ref wordsCount, size);

words[words.Length - 1] = temp[i];

wordsCount[words.Length - 1] = 1;

}

}

}

}

int temp1;

string temp2;

for (int i = 0; i < wordsCount.Length - 1; i++)

{

for (int j = i + 1; j < wordsCount.Length; j++)

{

if (wordsCount[i] < wordsCount[j])

{

temp1 = wordsCount[i];

wordsCount[i] = wordsCount[j];

wordsCount[j] = temp1;

temp2 = words[i];

words[i] = words[j];

words[j] = temp2;

}

}

}

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

{

Console.WriteLine($"{wordsCount[i]} | {words[i]} ");

}

Console.ReadKey();

}

**Результат работы программы:**



**Задание 2. Утилита nl выводит переданный файл в стандартный вывод или в другой файл, выполняя**

**нумерацию его строк. Если файл не задан или задан как –, читает стандартный ввод.**

**Формат использования: nl [-i] [-l] [-n] входной\_файл [выходной\_файл]**

**• -i ЧИСЛО Задает шаг увеличения номеров строк**

**• -l 1/0 Задает флаг нумерации пустых строк**

**• -n ФОРМАТ Использовать заданный формат для номеров строк.**

**ln – номер выравнен по левому краю, без начальных нулей**

**rn – номер выровнен по правому краю, без начальных нулей**

**rz – номер выровнен по правому краю с начальными нулями**

**Пример использования: nl -i 2 -l 0 -n ln in.txt**

**Обрабатывает файл in.txt , выводит результат в стандартный вывод, инкремент счетчика равен**

**двум (-i 2), пустые строки не нумеруются.**

**Реализация алгоритмов:**

static void Main(string[] args)

{

string directory1 = Directory.GetCurrentDirectory() + @"\";

string directory2 = Directory.GetCurrentDirectory() + @"\";

string firstFile = "";

string secondFile = "";

Console.WriteLine("Введите команду:");

string[] command = Console.ReadLine().Split(' ');

int iShag = 1, lNull = 0;

Alignment alignment = Alignment.left;

if (command.Length < 3)

return;

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

{

if (i == 0)

if (command[i] != "nl")

return;

if (command[i] == "-i")

{

i++;

iShag = Convert.ToInt32(command[i]);

}

if (command[i] == "-l")

{

i++;

lNull = Convert.ToInt32(command[i]);

}

if (command[i] == "-n")

{

i++;

switch (command[i])

{

case "ln":

alignment = Alignment.left;

break;

case "rn":

alignment = Alignment.right;

break;

case "rz":

alignment = Alignment.rightWithNull;

break;

default:

Console.WriteLine("something go wrong");

return;

}

}

if (i == command.Length - 2)

if (command[i].Contains(".txt"))

{

firstFile += command[i];

secondFile += command[++i];

}

else

{

firstFile += command[++i];

}

}

int count = 0;

int num = 1;

string[] words = new string[0];

using (StreamReader fs = new StreamReader($@"{directory1}{firstFile}"))

{

while (!fs.EndOfStream)

{

count++;

string temp = fs.ReadLine();

if (lNull == 0)

if (temp == "")

{

Array.Resize(ref words, count);

words[words.Length - 1] = temp;

continue;

}

Array.Resize(ref words, count);

words[words.Length - 1] = Numbering(temp, alignment, num);

num += iShag;

}

}

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

{

Console.WriteLine(words[i]);

}

if (secondFile != "")

{

using (StreamWriter sw = new StreamWriter($@"{directory2}{secondFile}"))

{

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

sw.WriteLine(words[i]);

}

}

Console.ReadKey();

}

static string Numbering(string str, Alignment alignment, int num)

{

switch (alignment)

{

case Alignment.left:

str = num + ". " + str;

break;

case Alignment.right:

str = str + " ." + num;

break;

case Alignment.rightWithNull:

string result = String.Format("{0:D10}", num);

str = str + " ." + result;

break;

}

return str;

}

enum Alignment

{

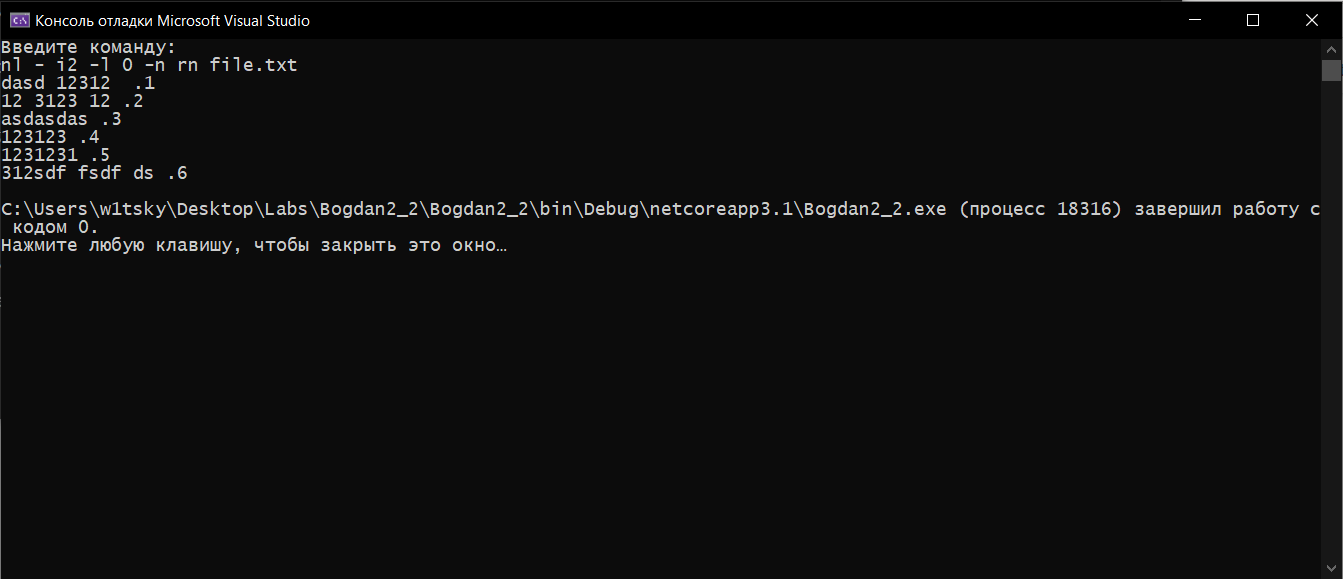
left,

right,

rightWithNull

}

**Результат работы программы:**

Вывод: приобрел практические навыки обработки параметров командной строки, закрепил базовые знания языка программирования C# при решении практических задач.