Министерство образования Республики Беларусь

Учреждение образования «Брестский государственный технический университет»

Кафедра ИИТ

Отчёт по лабораторной работе №2

Выполнил:

студент 3 курса

Группы ПО-5

Поздняков Д.А.

Проверил:

Крощенко А.А.

Брест 2021

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

**Вариант 5**

**Задание 1.** Напишите программу, которая cчитывает текст и печатает таблицу, показывающую, сколько раз в этом тексте встречаются однобуквенные слова, двухбуквенные слова, трехбуквенные слова и т.д.

**Задание 2.** Утилита cp осуществляет копирование файла из одного каталога в другой. Исходный файл остаётся неизменным, имя созданного файла может быть таким же, как у исходного, или измениться.

Формат использования: cp [-f ][-i][-n] исходный\_файл целевой\_файл

• -f Разрешает удаление целевого файла, в который производится копирование, если он не может быть открыт для записи.

• -i Утилита будет запрашивать, следует ли перезаписывать конечный файл, имя которого совпадает с именем исходного. Для того, чтобы перезаписать файл, следует ввести y или его эквивалент. Ввод любого другого символа приведёт к отмене перезаписи данного файла.

• -n Не перезаписывать существующий файл (отменяет предыдущий параметр - i ).

Пример использования: cp -fn src.txt dest.txt Копирует содержимое из src.txt в dest.txt с ключами -f и -n.

**Код программы (задание 1):**

using System;

using System.IO;

namespace lab2

{

class lab2

{

static void Main(string[] args)

{

string path = @"D:\repos\spp\lab2\SomeText.txt";

// чтение из файла

FileStream fstream = File.OpenRead(path);

// преобразуем строку в байты

byte[] array = new byte[fstream.Length];

// считываем данные

fstream.Read(array, 0, array.Length);

// декодируем байты в строку

string FileText = System.Text.Encoding.Default.GetString(array);

Console.WriteLine($"Текст из файла:\n{FileText}");

char[] separators = new char[] { ' ', '.', ',', '!', '?', '-', '(', ')' , '\n', '\t', '\"', ';', ':', '\r'};

string[] subs = FileText.Split(separators, StringSplitOptions.RemoveEmptyEntries);

int[] frequency = new int[15];

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

{

frequency[subs[i].Length]++;

}

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

{

Console.WriteLine($"{frequency[i]}\t {i}-буквенных");

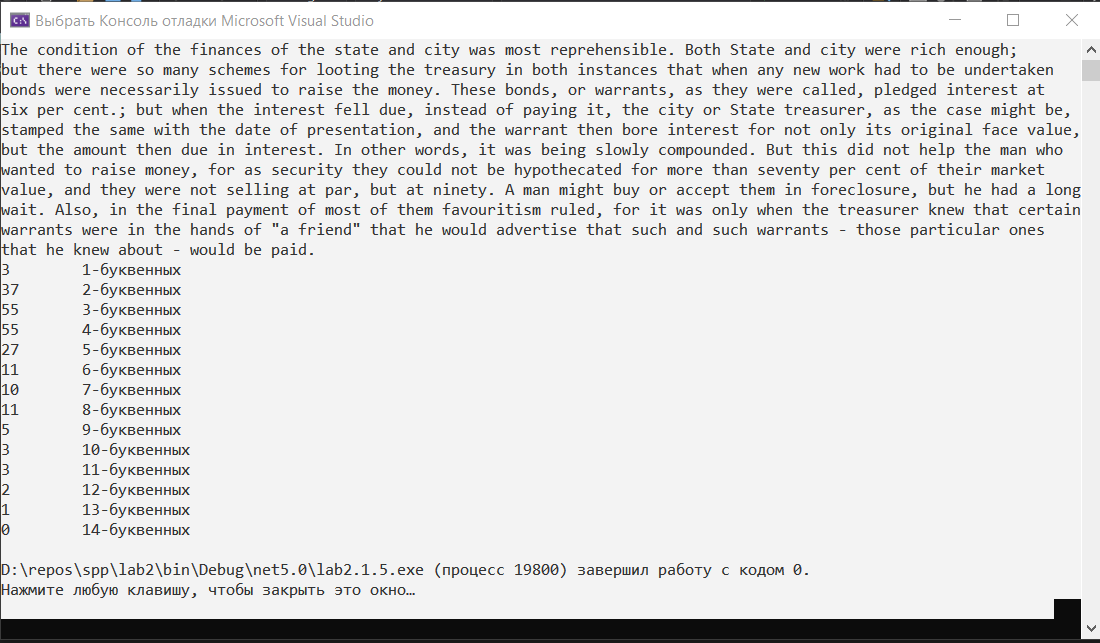
}

}

}

}

**Результат выполнения программы:**



**Код программы (задание 2):**

using System;

using System.IO;

using System.Diagnostics;

namespace lab2.\_2

{

class Program

{

static void Main(string[] args)

{

string str = "";

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

{

str += args[j];

}

char[] separators = new char[] { ' ' };

string[] subs = str.Split(separators, StringSplitOptions.RemoveEmptyEntries);

if (subs.Length != 4)

{

Console.WriteLine("Неверный формат ввода!1");

Process.GetCurrentProcess().Kill();

}

if (subs[0] != "cp")

{

Console.WriteLine("Неверный формат ввода!2");

Process.GetCurrentProcess().Kill();

}

char[] temp = subs[1].ToCharArray();

if (temp[0] != '-')

{

Console.WriteLine("Неверный формат ввода!3");

Process.GetCurrentProcess().Kill();

}

if (temp.Length > 4)

{

Console.WriteLine("Неверный формат ввода!4");

Process.GetCurrentProcess().Kill();

}

int f = 0, i = 0, n = 0;

bool[] fin = new bool[] { false, false, false };

foreach (var A in temp)

{

switch (A)

{

case 'f':

f++;

if (f > 1)

{

Console.WriteLine("Неверный формат ввода!5");

Process.GetCurrentProcess().Kill();

}

fin[0] = true;

if (fin[2] == true)

{

fin[1] = false;

}

break;

case 'i':

i++;

if (i > 1)

{

Console.WriteLine("Неверный формат ввода!6");

Process.GetCurrentProcess().Kill();

}

fin[1] = true;

if (fin[2] == true)

{

fin[1] = false;

}

break;

case 'n':

n++;

if (n > 1)

{

Console.WriteLine("Неверный формат ввода!7");

Process.GetCurrentProcess().Kill();

}

fin[2] = true;

if (fin[2] == true)

{

fin[1] = false;

}

break;

}

}

if (fin[0] == true && fin[2] == true)

{

//Console.WriteLine("fn");

FileStream file = File.OpenWrite(subs[3]);

FileInfo iFile = new FileInfo(subs[3]);

if (file == null)

{

file.Close();

iFile.Delete();

System.IO.File.Create(subs[3]);

}

file.Close();

CopyWithoutRewriting(subs);

Process.GetCurrentProcess().Kill();

}

if (fin[0] == true && fin[1] == true)

{

//Console.WriteLine("fi");

FileStream file = File.OpenWrite(subs[3]);

FileInfo iFile = new FileInfo(subs[3]);

if (file == null)

{

file.Close();

iFile.Delete();

System.IO.File.Create(subs[3]);

}

file.Close();

Console.WriteLine("Перезаписать целевой файл? (y/n):");

string YorN = Console.ReadLine();

if (YorN == "y" || YorN == "Y" || YorN == "yes" || YorN == "Yes")

{

Copy(subs);

}

else

{

CopyWithoutRewriting(subs);

}

Process.GetCurrentProcess().Kill();

}

if (fin[2] == true)

{

//Console.WriteLine("n");

CopyWithoutRewriting(subs);

Process.GetCurrentProcess().Kill();

}

if (fin[1] == true)

{

//Console.WriteLine("i");

Console.WriteLine("Перезаписать целевой файл? (y/n):");

string YorN = Console.ReadLine();

if(YorN == "y" || YorN == "Y" || YorN == "yes" || YorN == "Yes")

{

Copy(subs);

}

else

{

CopyWithoutRewriting(subs);

}

Process.GetCurrentProcess().Kill();

}

if (fin[0] == true)

{

//Console.WriteLine("f");

FileStream file = File.OpenWrite(subs[3]);

FileInfo iFile = new FileInfo(subs[3]);

if (file == null)

{

file.Close();

iFile.Delete();

System.IO.File.Create(subs[3]);

}

file.Close();

Copy(subs);

Process.GetCurrentProcess().Kill();

}

}

static void Copy(string[] subs)

{

FileStream fstream = File.OpenRead(subs[2]);

byte[] array = new byte[fstream.Length];

fstream.Read(array, 0, array.Length);

using (FileStream fstream1 = new FileStream(subs[3], FileMode.OpenOrCreate))

fstream1.Write(array, 0, array.Length);

}

static void CopyWithoutRewriting(string[] subs)

{

FileStream fstream = File.OpenRead(subs[2]);

byte[] array = new byte[fstream.Length];

fstream.Read(array, 0, array.Length);

using (FileStream fstream1 = new FileStream(subs[3], FileMode.Append))

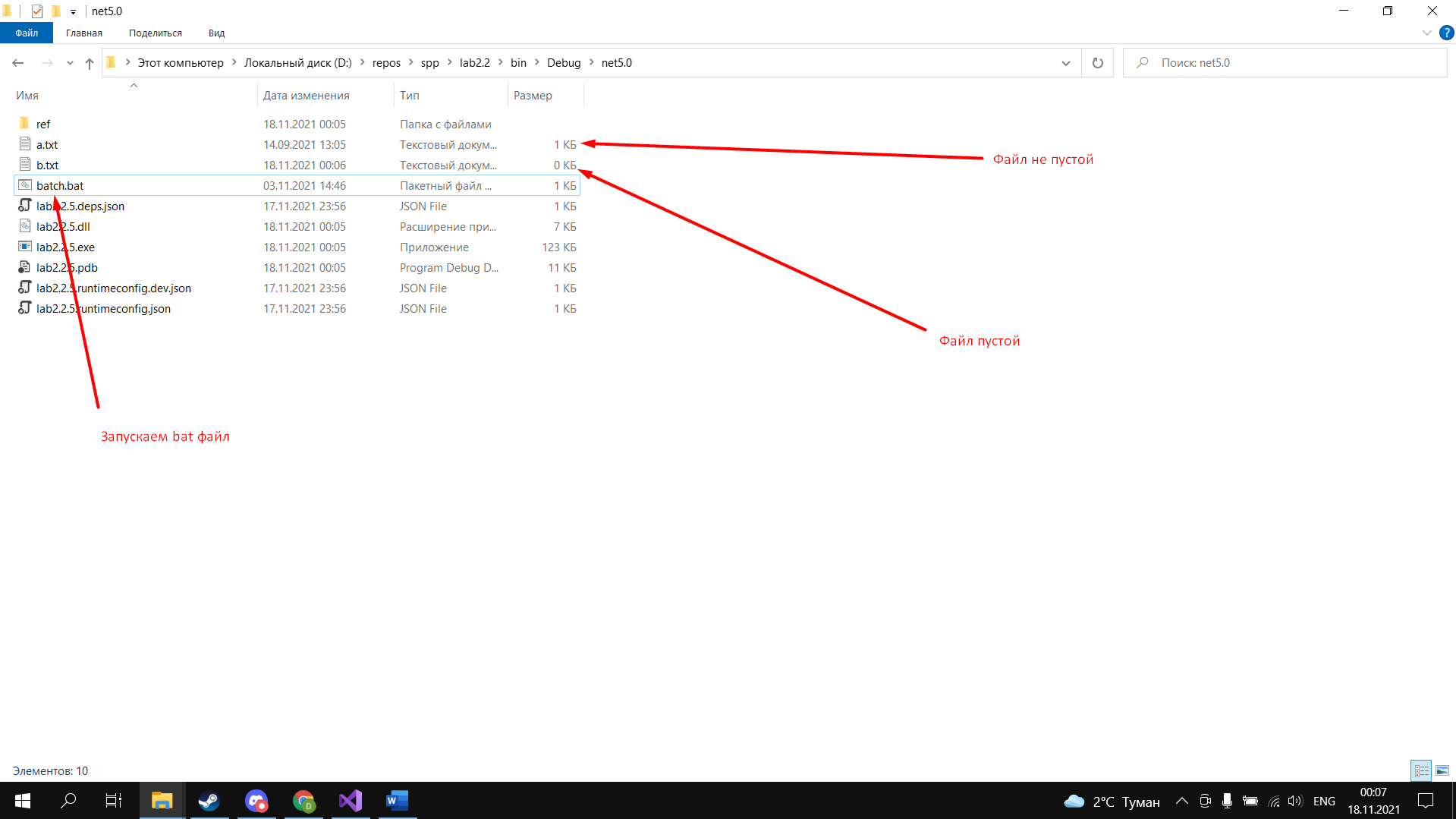
fstream1.Write(array, 0, array.Length);

}

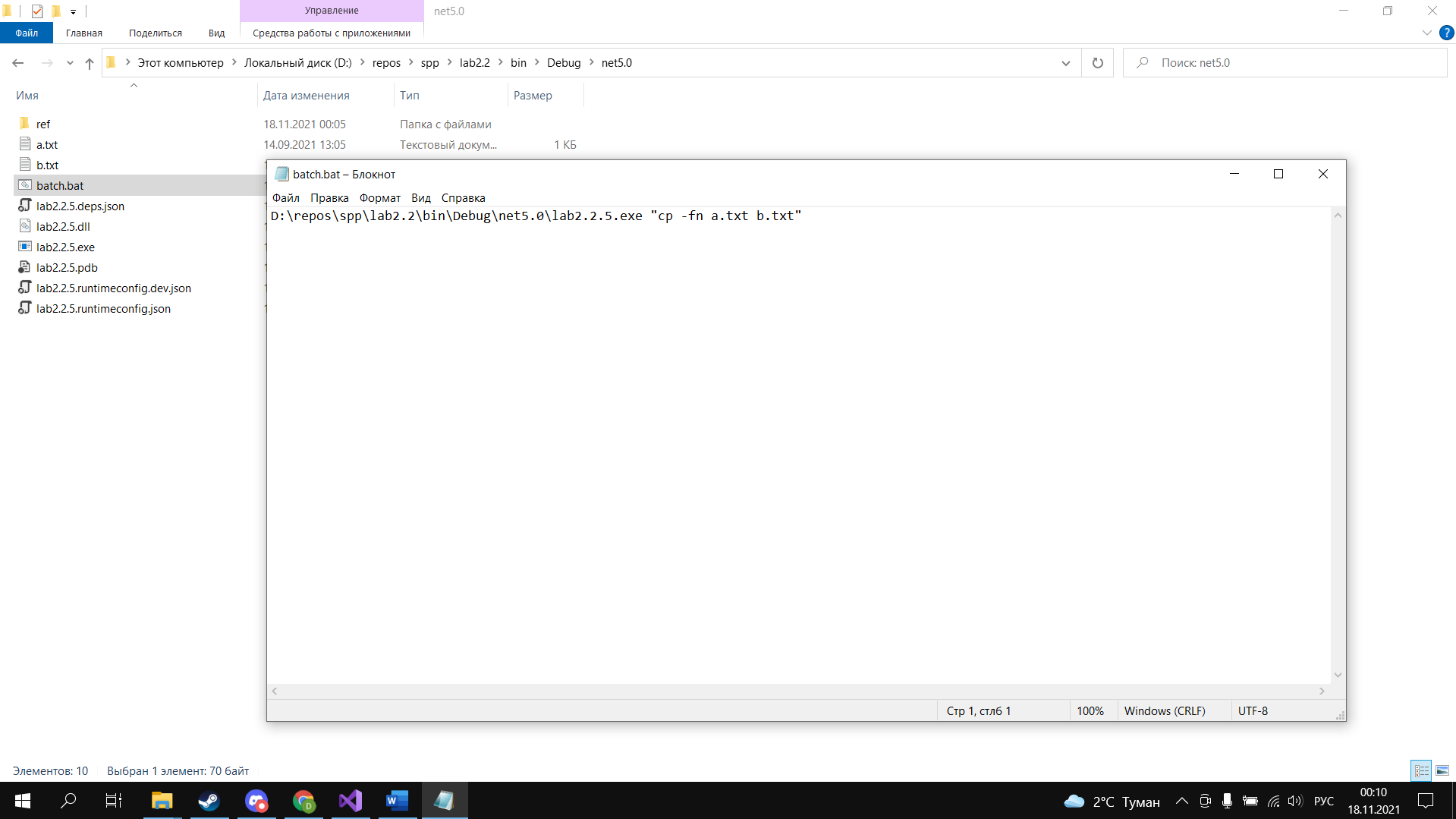
}

}

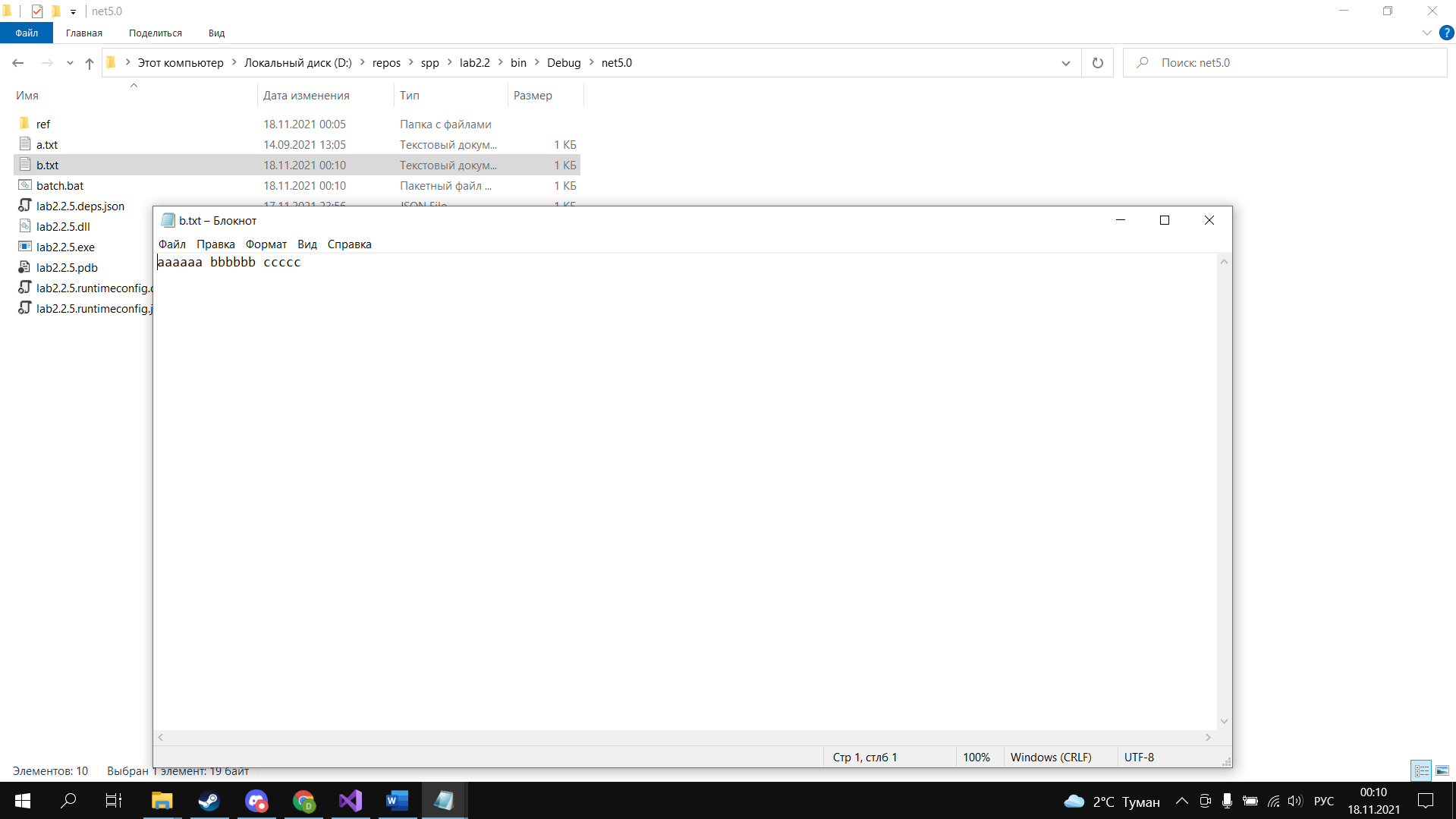
**Результат выполнения программы:**



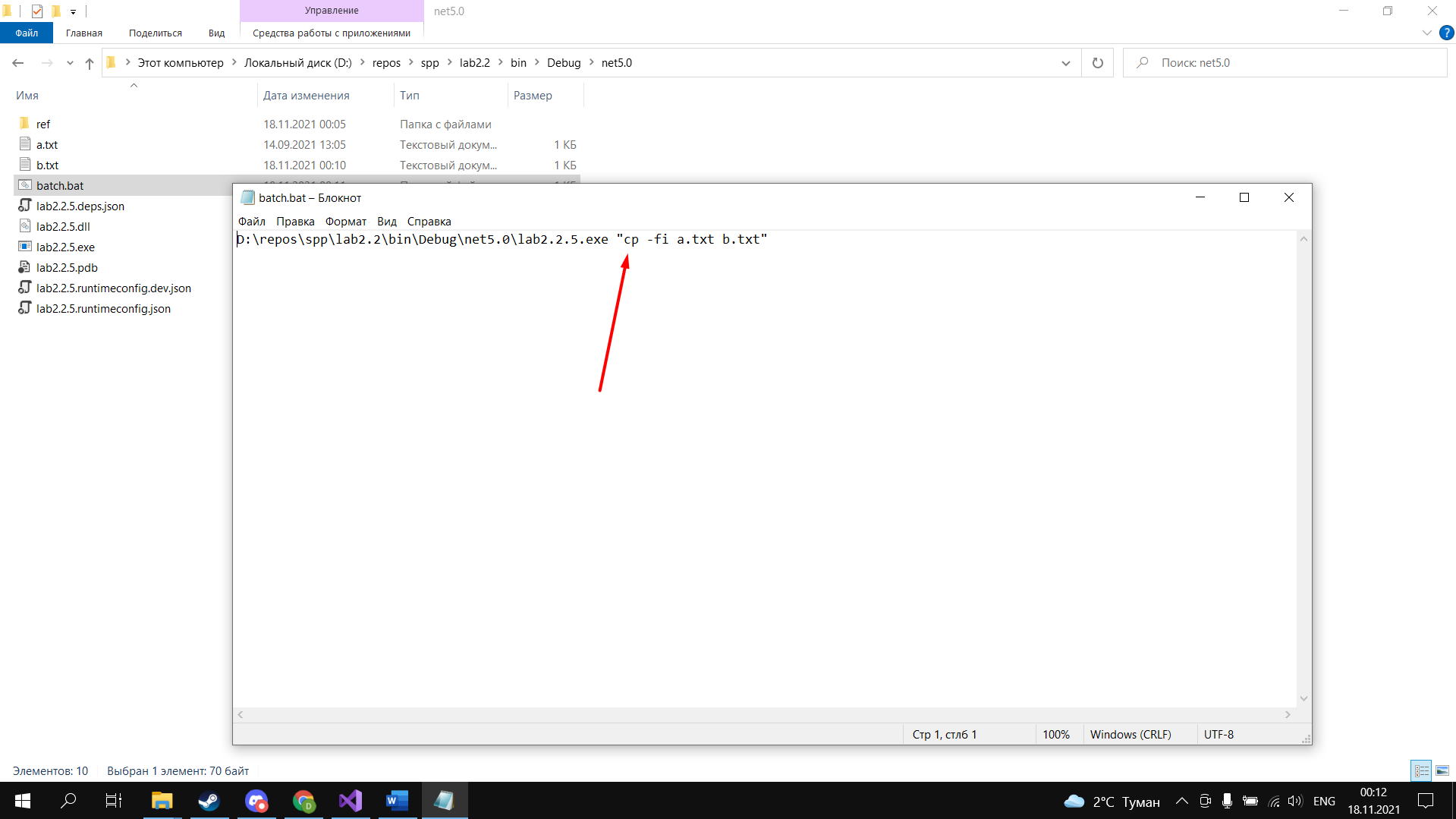
Содержимое bat файла:



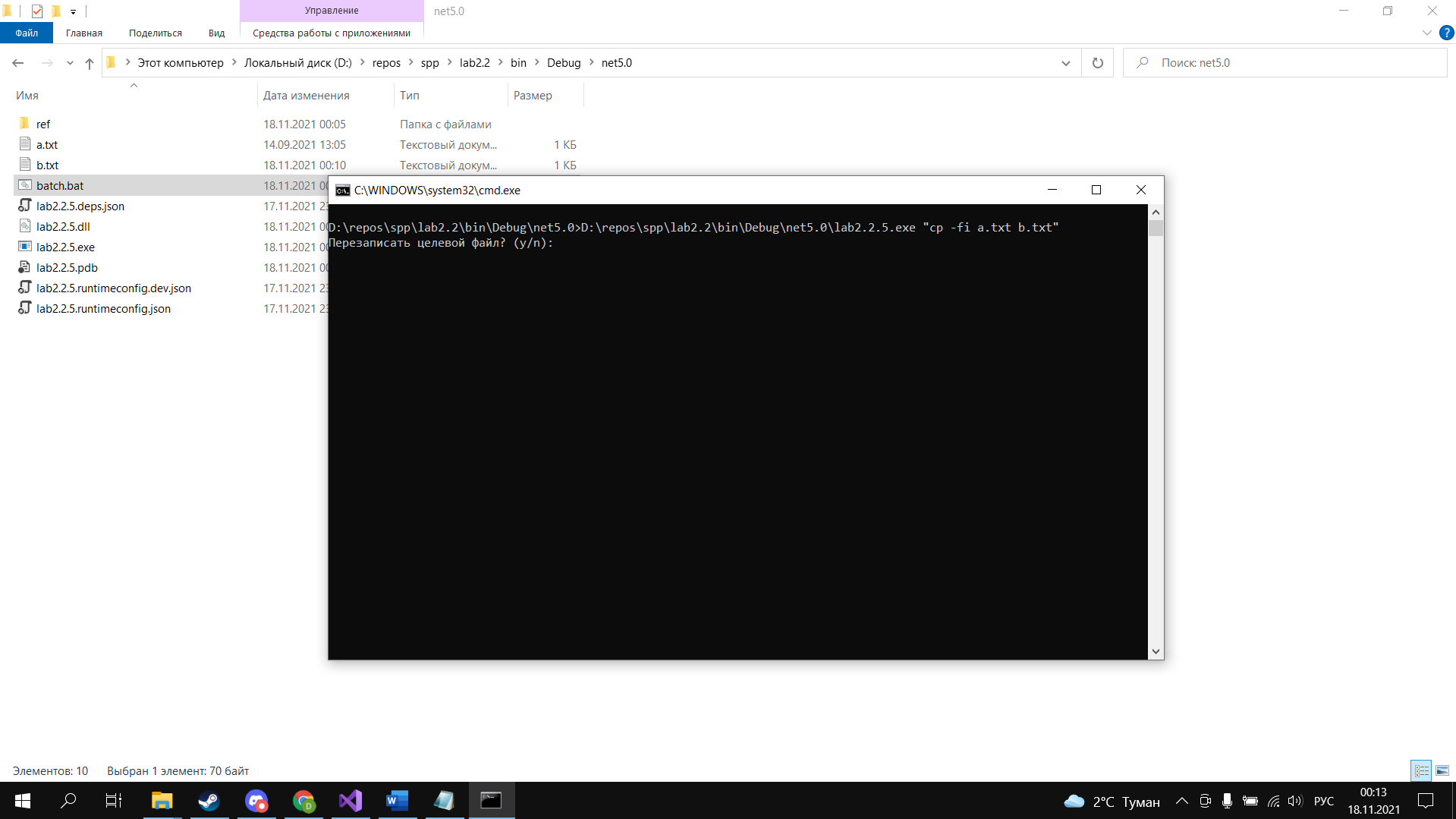
Результат после запуска файла:



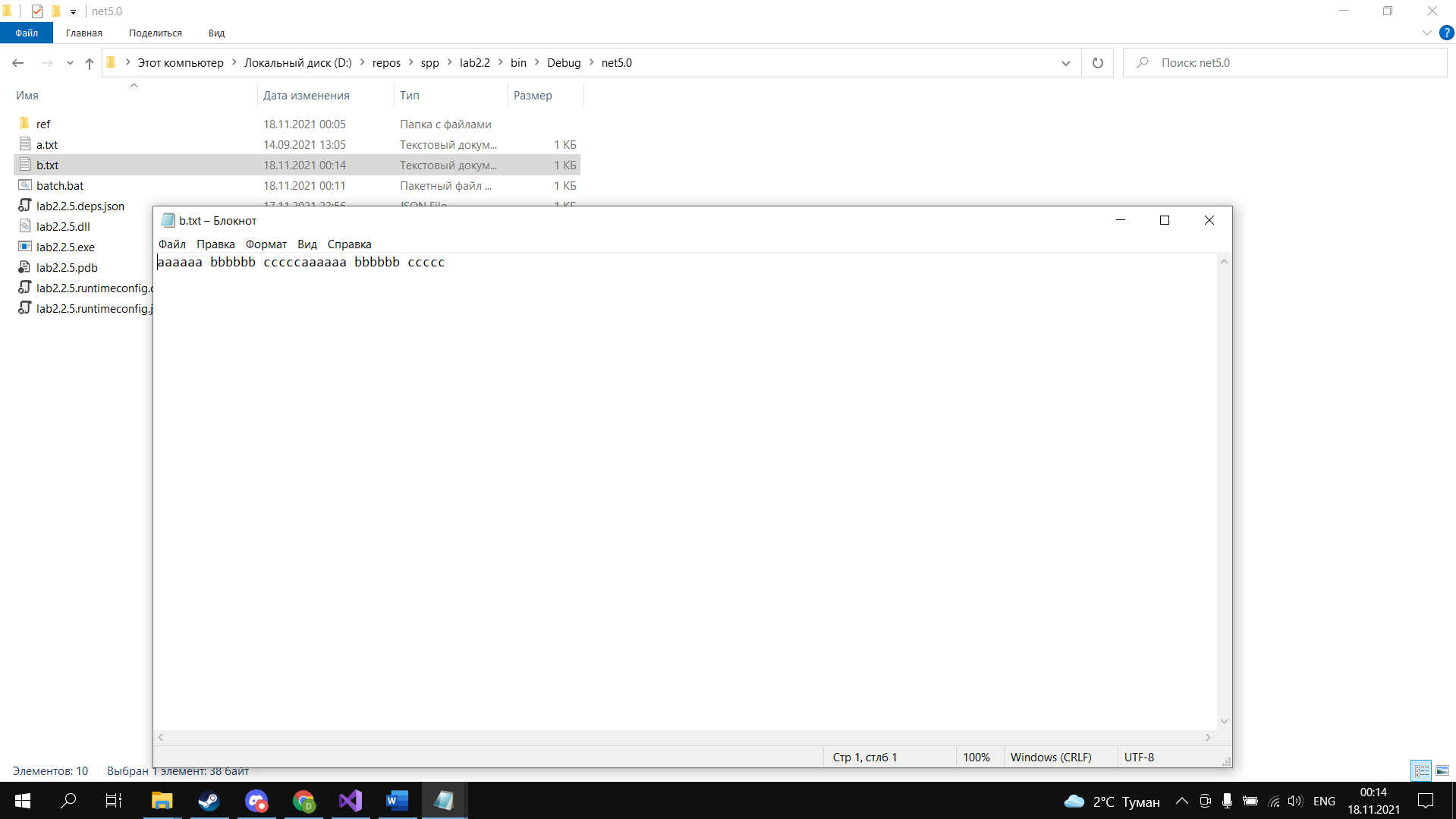
Изменяем ключ в bat файле:



Запускаем bat файл:



У нас спрашиваю стоит ли перезаписать целевой файл, отвечаем нет и ожидаем что в файле b.txt станет больше текста



Так оно и есть, в файле остались данные из предыдущего копирования и записались новые данные.

Вывод: в ходе выполнения лабораторной работы приобрел базовые навыки работы с файловой системой на C#.