Учреждение образования

«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет информационных технологий

Кафедра информационных систем и технологий

**Лабораторная работа № 2**

«Элементы теории информации. Информативность данных в различных кодировках»

Выполнил:

Студент: Герман А.Е.

ФИТ 3 курса 5 группы

Преподаватель: Савельева М.Г.

Минск 2024

# **1 Описание приложения**

Данное консольное приложение, разработанное на языке программирования C#, предназначено для анализа текстового файла и конвертации его содержимого в формат base64, а также расчёта энтропии Шеннона и Хартли, а также избыточность алфавита.

Основные функциональности приложения:

* Чтение текстового файла;
* Конвертация содержимого текстового файла в формат base64;
* Анализ алфавитов в текстовом документе;
* Подсчёт энтропии и избыточности алфавитов.

Конвертация и проведение операций с буфферами:

* Уравнивание длин буфферов для корректного проведения операций;
* Проведение операции XOR.

# **2 Методика выполнения расчетов**

В данной лабораторной работе была поставлена цель создания приложения, позволяющее конвертировать текст и работать с буфферами. На листингах 2.1 и 2.2 представлен классы, реализующий данную функциональность.

|  |
| --- |
| using System;  using System.Collections.Generic;  using System.IO;  namespace is\_lab\_2  {  class TextAnalyzer  {  private string text;  public TextAnalyzer(string text)  {  this.text = text;  }  public void Analyze()  {  Dictionary<char, int> charFrequency = new Dictionary<char, int>();  foreach (char c in text)  {  if (char.IsLetter(c))  {  if (charFrequency.ContainsKey(c))  charFrequency[c]++;  else  charFrequency[c] = 1;  }  }  Console.WriteLine("Частоты букв в тексте:");  foreach (var pair in charFrequency)  {  Console.WriteLine($"{pair.Key}: {pair.Value}");  }  int alphabetSize = charFrequency.Count;  double hartleyEntropy = Math.Log2(alphabetSize);  double shannonEntropy = 0;  foreach (var pair in charFrequency)  {  double probability = (double)pair.Value / text.Length;  shannonEntropy -= probability \* Math.Log2(probability);  }  double redundancy = hartleyEntropy - shannonEntropy;  Console.WriteLine($"Энтропия Хартли: {hartleyEntropy}");  Console.WriteLine($"Энтропия Шеннона: {shannonEntropy}");  Console.WriteLine($"Избыточность алфавита: {redundancy}");  }  }  } |

Листинг 2.1 – Класс **TextAnalyzer**, реализующий функционал для работы с текстом

|  |
| --- |
| using System;  namespace is\_lab\_2  {  class BufferProcessor  {  public static string PadAndXOR(string bufferA, string bufferB, InputFormat formatA, InputFormat formatB)  {  byte[] bytesA = ConvertInputToBytes(bufferA, formatA);  byte[] bytesB = ConvertInputToBytes(bufferB, formatB);  int maxLength = Math.Max(bytesA.Length, bytesB.Length);  bytesA = PadBuffer(bytesA, maxLength);  bytesB = PadBuffer(bytesB, maxLength);  byte[] resultBytes = new byte[maxLength];  for (int i = 0; i < maxLength; i++)  {  resultBytes[i] = (byte)(bytesA[i] ^ bytesB[i]);  }  string result = ConvertBytesToString(resultBytes, formatA);  return result;  }  private static byte[] ConvertInputToBytes(string input, InputFormat format)  {  return format switch  {  InputFormat.ASCII => System.Text.Encoding.ASCII.GetBytes(input),  InputFormat.Base64 => Convert.FromBase64String(input),  \_ => throw new ArgumentException("Unsupported input format"),  };  }  public enum InputFormat  {  ASCII,  Base64  }  }  } |

Листинг 2.2 – Класс **BufferProcessor**, реализующий функционал для работы с буфферами

Метод **Analyze** используется для получения энтропии для текста алфавита и избыточности алфавита. Метод **PadAndXOR** осуществляет операцию XOR над двумя строками.

# **3 Результаты работы приложения**

Для выполнения расчетов достаточно необходимо запустить приложение. Рисунки 3.1 - 3.3 показывают необходимые расчеты и вызовы методов, требуемые в данной лабораторной работе.

## **3.1 Вывод частоты алфавита и его избыточности**

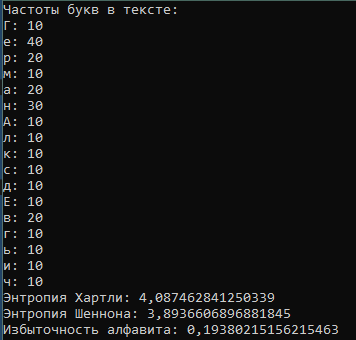


Рисунок 3.1 – Результат работы метода **Analyze** с входного файла(ASCII)

В качестве тестового текста было использовано ФИО. Используя формулу Шеннона и Хартли, было найдено значение избыточности алфавита – 0,1938.

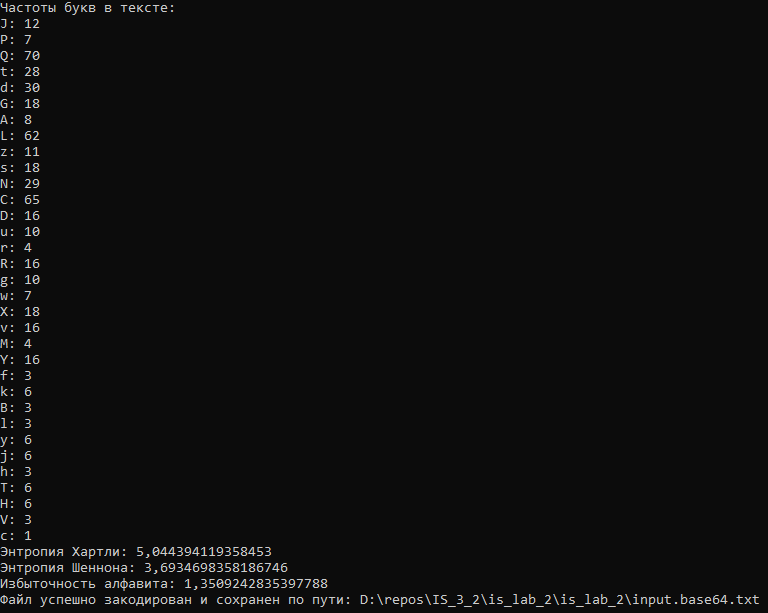


Рисунок 3.2 – Результат работы метода **Analyze** для выходящего файла(base64)

В данном примере избыточность алфавита равна 1,350924. Это означает, что разница между алфавитами значительна: большее число символов, большая частота использования символов, но из-за расширившегося пула символов вероятность выпадения символов стратифицировалась из-за чего энтропия Шеннона понизилась.

## **3.2 Выполнение операции над буфферами**

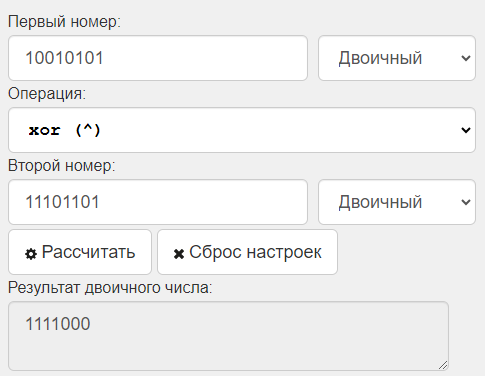


Рисунок 3.3 – Результат работы метода **PadAndXOR**

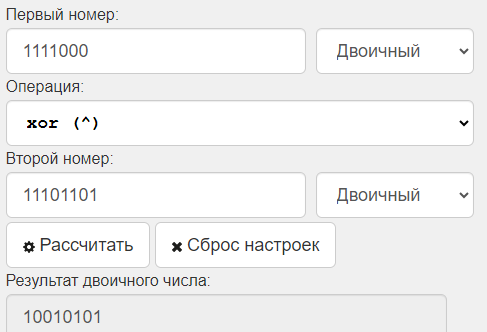
Достижение необходимой операции *a*XOR*b*XOR*b* выполняется путём рекурсивного вызова функции. Результат преобразования соответствует *a* – фамилии, так как дважды использовалась операция исключающего или(**Xor**) с одним и тем же вторым операндом.

Пример: *a* = 10010101, *b* = 11101101:

1. 10010101 Xor 11101101 = 1111000



1. 1111000 Xor 11101101 = 10010101



Что и требовалось доказать, на выходе получилось исходное *a* **=** 10010101.

## **3.3 Гистограммы**

Рисунок 3.4 – Гистограмма для входящего алфавита

Рисунок 3.5 – Гистограмма для исходящего алфавита

# **4 Вывод**

В ходе изучения теоретических материалов лабораторной работы и выполнения её практической части были изучены основы теории информации и анализа энтропии. Это консольное приложение, разработанное на языке программирования C#, предоставляет функциональности, направленные на измерение энтропии в различных сценариях, связанных с текстовыми данными.