Учреждение образования

«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Лабораторная работа №2

**Элементы теории информации. Параметры и характеристики дискретных информационных систем**

Студент: Чёрная Я.Р.

ФИТ 3 курс 4 группа

Преподаватель: Нистюк О.А.

Минск 2025

**Задания 1/2:**

В функции ReadFromFile, происходит чтение текста из файла. Текст приводится к нижнему регистру для унификации. Это необходимо, чтобы избежать учета символов в разных регистрах как различных, так как для расчета энтропии важно учитывать все повторения символов одинаково.

|  |
| --- |
| public static string ReadFromFile(string filePath)  {  if (!File.Exists(filePath)) { return ""; }  return File.ReadAllText(filePath, Encoding.UTF8).ToLower();  } |

Функция GetSymbolFrequencies анализирует текст и подсчитывает, сколько раз каждый символ появляется в нем.

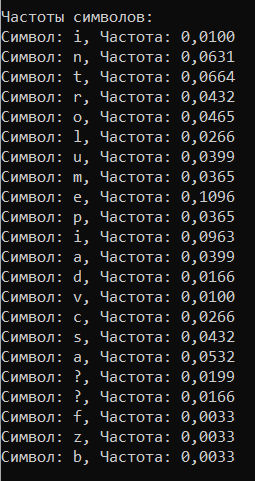
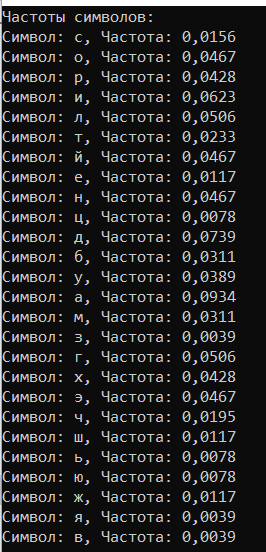
|  |
| --- |
| public static Dictionary<char, int> GetSymbolFrequencies(string text)  {  Dictionary<char, int> frequencies = new Dictionary<char, int>();  foreach (char c in text)  {  if (char.IsLetterOrDigit(c))  { if (frequencies.ContainsKey(c)) frequencies[c]++;  else frequencies[c] = 1;  }  }  return frequencies;  } |

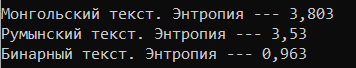
ВGetSymbolProbabilities рассчитываем вероятность появления каждого символа в тексте. Для этого частота каждого символа делится на общую длину текста.

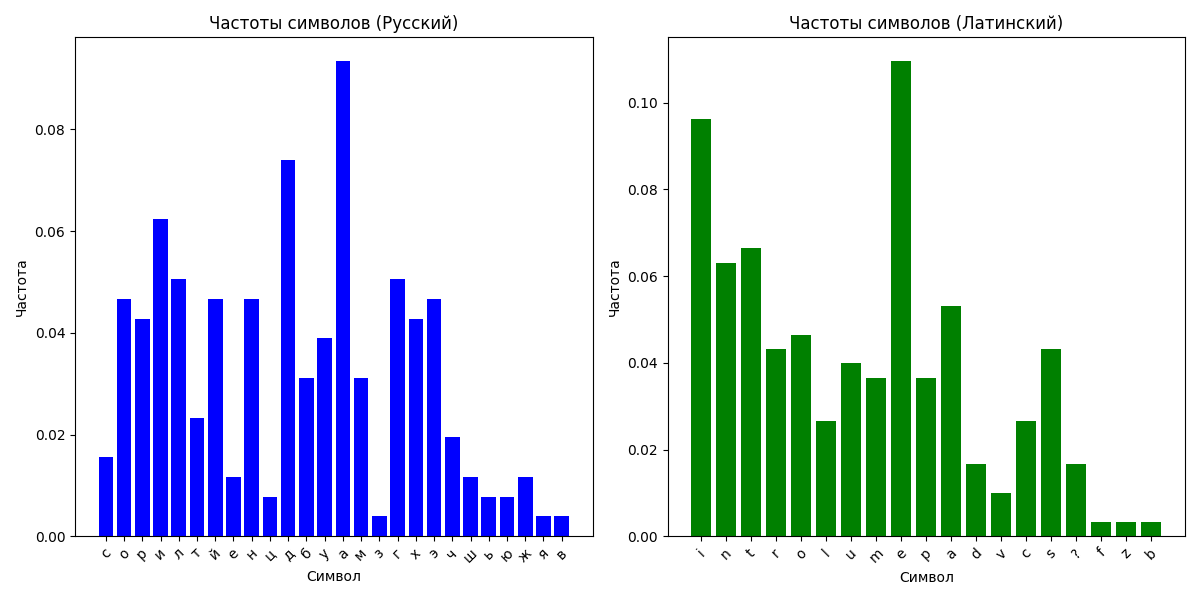
|  |
| --- |
| public static Dictionary<char, double> GetSymbolProbabilities(string text)  {  var frequencies = GetSymbolFrequencies(text);  double totalCount = text.Length;  Dictionary<char, double> probabilities = frequencies.ToDictionary(  pair => pair.Key,  pair => pair.Value / totalCount);  return probabilities;  } |

Используем формулу Шеннона для расчета энтропии текста.

|  |
| --- |
| public static double GetShannonEntropy(string text)  {  var probabilities = GetSymbolProbabilities(text);  double entropy = 0;  foreach (var prob in probabilities.Values)  { if (prob > 0) entropy -= prob \* Math.Log2(prob); }  return Math.Round(entropy, 3);  } |



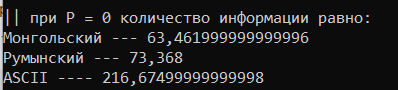




**Задание 3:**

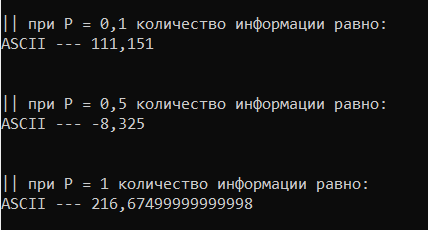
Найти количество информации в фамилии, имени и отчестве, используя формулу:

|  |
| --- |
| public static double GetInformationAmount(string text)  {  double entropy = GetShannonEntropy(text);  return entropy \* text.Length;  } |



В ASCII каждый символ кодируется 8 битами, даже если он встречается часто и можно было бы использовать более короткий код. Поэтому данный алфавит избыточен. Чем больше размер алфавита и фиксированная длина кода, тем больше избыточности и выше энтропия.

**Задание 4:**



Чем больше вероятность ошибки, тем меньше полезной информации передается. При P = 0.5 информация почти исчезает, а при P = 1 сообщение становится случайным шумом.

**Вывод**

В ходе лабораторной работы были закреплены теоретические знания по основам теории информации.Также было разработано приложение для расчета и анализа параметров и информативных характеристик дискретных ИС.