Учреждение образования

«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Защита информации и надёжность информационных систем

Студент: Лопатнюк П.В.

ФИТ 3 курс 1 группа

Преподаватель: Нистюк О.А.

Минск 2025

**Лабораторная работа № 2**

**ЭЛЕМЕНТЫ ТЕОРИИ ИНФОРМАЦИИ. ПАРАМЕТРЫ И ХАРАКТЕРИСТИКИ ДИСКРЕТНЫХ ИНФОРМАЦИОННЫХ СИСТЕМ**

**Цель:** приобретение практических навыков расчета и анализа параметров и информативных характеристик дискретных ИС.

**Задачи:**

1. Закрепить теоретические знания по основам теории информации.

2. Разработать приложение для расчета и анализа параметров и информативных характеристик дискретных ИС.

3. Результаты выполнения лабораторной работы оформить в виде описания разработанного приложения, методики выполнения экспериментов с использованием приложения и результатов эксперимента.

# Теоретические сведения

Передача информации (данных) осуществляется между двумя абонентами, называемыми источником сообщения (ИсС) и получателем сообщения (ПС). Третьим элементом информационной системы является канал (среда) передачи, связывающий ИсС и ПС.

Отметим также, что и в системах с хранением информации всегда можно выделить ИсС и ПС. В данном случае каналом передачи выступает устройство хранения информации (память). Например, при записи данных в ОЗУ (оперативное запоминающее устройство) компьютера в качестве ИсС и ПС может выступать процессор (соответственно при записи и чтении данных).

Таким образом, простейшая информационная система состоит из трёх элементов: источника сообщения, канала передачи сообщения и получателя сообщения.

Отображение сообщения обеспечивается изменением какой-либо физической величины, характеризующей процесс (например, амплитуда, частота, фаза). Эта величина является информационным параметром сигнала (в общем случае — информационной системы).

Сигналы, как и сообщения, могут быть непрерывными и дискретными. Информационный параметр непрерывного сигнала с течением времени может принимать любые мгновенные значения в определённых пределах. Непрерывный сигнал часто называют аналоговым, а каналы и устройства, функционирующие на основе такого типа сигналов — аналоговыми.

Дискретный сигнал (устройство или канал передачи) характеризуется конечным числом значений информационного параметра. Дискретные сообщения состоят из последовательности дискретных знаков. Часто этот параметр принимает всего два значения — 0 или 1. Сообщение или канал его передачи на основе этих двух значений сигнала называют двоичным или бинарным.

Построение сигнала по определённым правилам, обеспечивающим соответствие между сообщением и сигналом, называют кодированием.

Кодирование в широком смысле — преобразование сообщения в сигнал.

Кодирование в узком смысле — представление исходных знаков, называемых символами, в другом алфавите с меньшим числом знаков. Оно осуществляется с целью повышения надёжности и преобразования сигналов к виду, удобному для передачи по каналам связи.

# Практическое задание

**Вариант 10:** Датский, Казахский

Создать приложение для расчета и анализа параметров и информативных характеристик дискретных ИС, с помощью которого:

а) рассчитать энтропию указанных преподавателем алфавитов: один – на латинице, другой – на кириллице (по формуле (2.1) перейти от частоты появления каждого символа алфавита к соответствующей вероятности); в качестве входного может быть принят произвольный электронный текстовый документ на основе соответствующего алфавита; частоты появления символов алфавитов оформить в виде гистограмм (можно воспользоваться приложением MS Excel);

Создаём файлы, в которых содержится большой произвольный текст на данных языках.

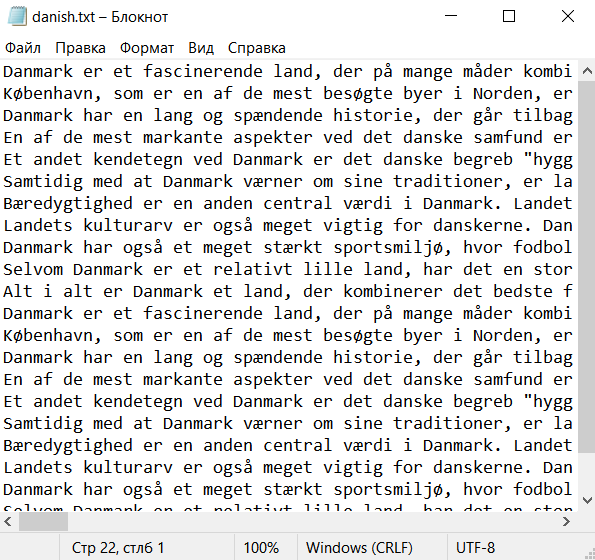
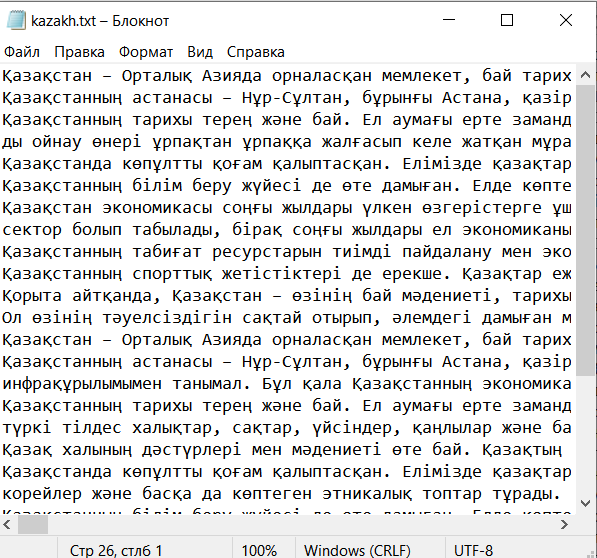


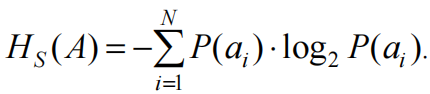
Рисунок 1.1 – Оригинальные тексты

Далее прописываем код для считывания текста из файлов:

|  |
| --- |
| public static string ReadTextFromFile(string filePath)  {  try  {  FileInfo fileInfo = new FileInfo(filePath);  if (fileInfo.Length < 10 \* 1024)  {  throw new Exception("Ошибка: файл слишком мал (меньше 10 КБ). Добавьте больше данных");  }  return File.ReadAllText(filePath);  }  catch (Exception ex)  {  Console.WriteLine($"Ошибка при чтении файла: {ex.Message}");  return string.Empty;  }  } |

Листинг 1.1 – Считывание элементов из файла

Энтропии алфавита А = {ai} по К. Шеннону рассчитывают по следующей формуле:



Пропишем следующую функцию для вычисления энтропии алфавита:

|  |
| --- |
| public static double CalculateEntropy(string text, char[] alphabet, string filePath)  {  text = new string(text.ToLower().Where(c => alphabet.Contains(c)).ToArray());  int textLength = text.Length;  if (textLength < 100)  {  Console.WriteLine("Текст слишком маленький для того, чтобы рассчитать энтропию");  return 0;  }  var frequency = new Dictionary<char, int>();  foreach (var letter in alphabet) frequency[letter] = 0;  foreach (var c in text) frequency[c]++;  Console.WriteLine("Частота появления символов:");  foreach (var kvp in frequency)  {  double probability = (double)kvp.Value / textLength;  Console.WriteLine($"Символ: '{kvp.Key}' Частота: {kvp.Value}, Вероятность: {probability:F4}");  }  SaveToExcel(frequency, textLength, filePath);  double entropy = 0;  foreach (var kvp in frequency)  {  double probability = (double)kvp.Value / textLength;  if (probability > 0) entropy += probability \* Math.Log2(probability);  }  return -entropy;  } |

Листинг 1.2 – Вычисление энтропии

Для наглядности запишем функцию для вывода данных частоты появления символов алфавитов:

|  |
| --- |
| private static void SaveToExcel(Dictionary<char, int> frequency, int textLength, string filePath)  {  using (var package = new ExcelPackage())  {  var worksheet = package.Workbook.Worksheets.Add("Data");  worksheet.Cells[1, 1].Value = "Символ";  worksheet.Cells[1, 2].Value = "Частота";  worksheet.Cells[1, 3].Value = "Вероятность";  int row = 2;  foreach (var kvp in frequency)  {  double probability = (double)kvp.Value / textLength;  worksheet.Cells[row, 1].Value = kvp.Key;  worksheet.Cells[row, 2].Value = kvp.Value;  worksheet.Cells[row, 3].Value = probability;  row++;  }  package.SaveAs(new System.IO.FileInfo(filePath));  }  } |

Листинг 1.3 – Метод сохранения в Excel

В итоге получаем следующую информацию:

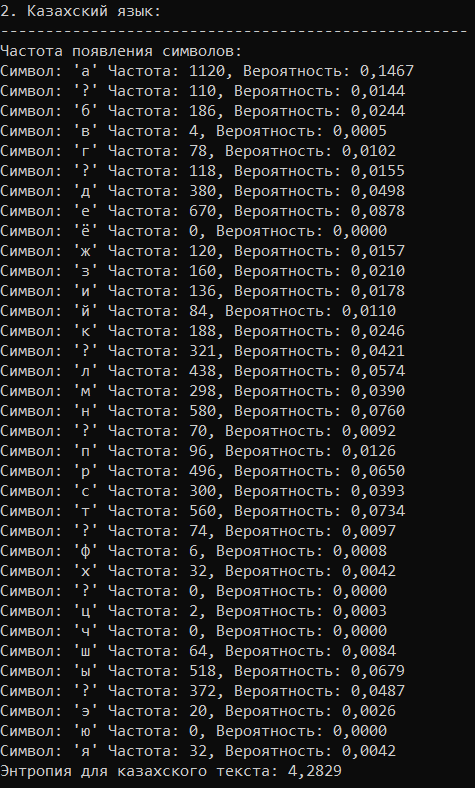
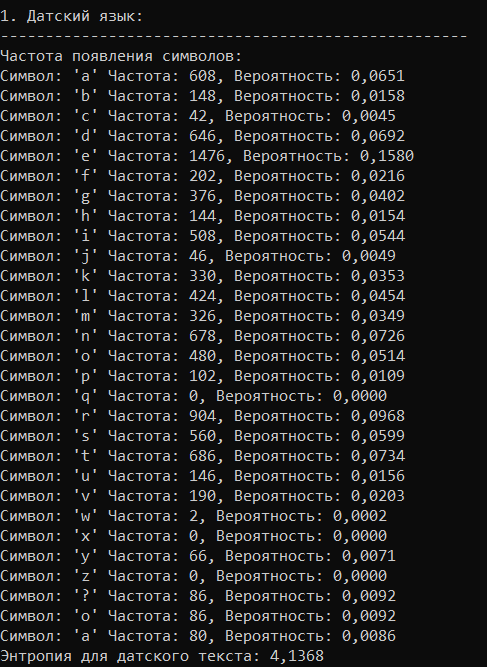


Рисунок 1.2 – Вероятности появления символов



Рисунок 1.3 – График появления каждого символа датского алфавита



Рисунок 1.4 – График появления каждого символа казахского алфавита

б) для входных документов, представленных в бинарных кодах, определить энтропию бинарного алфавита;

Записываем документы в бинарном виде:

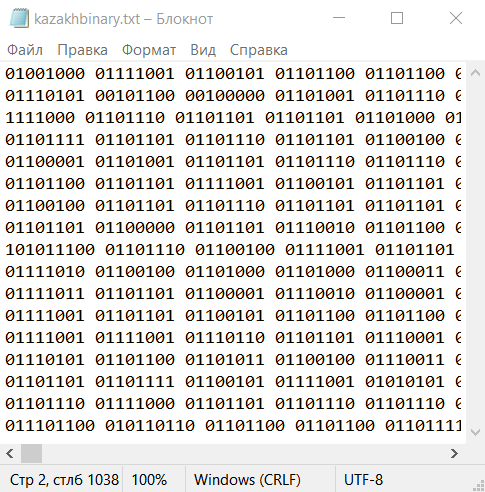
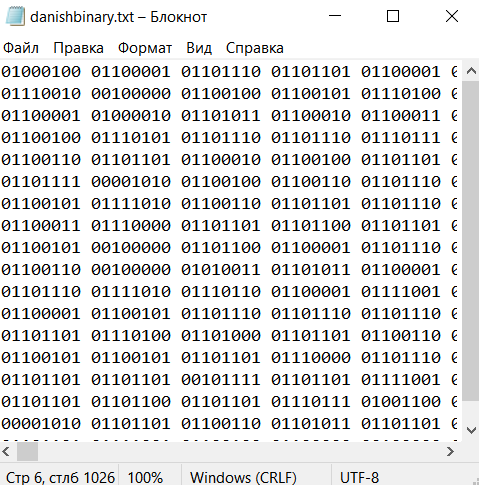


Рисунок 1.5 – Документы в бинарном виде

Используя ранее написанные функции получаем:

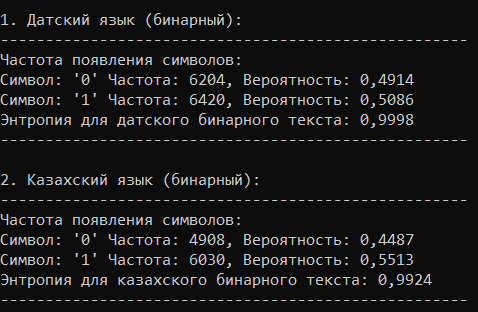


Рисунок 1.6 – Данные каждого алфавита

в) используя значения энтропии алфавитов, полученных в пунктах (а) и (б), подсчитать количество информации в сообщении, состоящем из собственных фамилии, имени и отчества (на основе исходного алфавита – (а) и в кодах ASCII – (б)); объяснить полученный результат;

Запишем функцию:

|  |
| --- |
| public static double CalculateInformationAmount(string text, double entropy, char[] alphabet)  {  text = new string(text.ToLower().Where(c => alphabet.Contains(c)).ToArray());  return entropy \* text.Length;  } |

Листинг 1.4 – Вычисление количества информации

В основном программном файле:

|  |
| --- |
| string danishFIO = "Lopatniuk Polina Vecheslavovna";  string kazakhFIO = "Лопатнюк Полина Вечеславовна";  string danishFIOBinary = "01010110 01101111 01100100 01100011 01101000 01111001 01110100 01110011 00100000 01000001 01101110 01100001 01110011 01110100 01100001 01110011 01101001 01111001 01100001 00100000 01010110 01101001 01110100 01100001 01101100 01101001 01100101 01110110 01101110 01100001";  string kazakhFIOBinary = "11010000 10010010 11010000 10111110 11010000 10110100 11010001 10000111 11010000 10111000 11010001 10000110 00100000 11010000 10010000 11010000 10111101 11010000 10110000 11010001 10000001 11010001 10000010 11010000 10110000 11010001 10000001 11010000 10111000 11010001 10001111 00100000 11010000 10010010 11010000 10111000 11010001 10000010 11010000 10110000 11010000 10111011 11010000 10111000 11010000 10110101 11010000 10110010 11010000 10111101 11010000 10110000";  double danishInformationAmount = EntropyCalculator.CalculateInformationAmount(danishFIO, danishEntropy, DanishAlphabet);  double kazakhInformationAmount = EntropyCalculator.CalculateInformationAmount(kazakhFIO, kazakhEntropy, KazakhAlphabet);  double danishInformationAmountBinary = EntropyCalculator.CalculateInformationAmount(danishFIOBinary, danishEntropyBinary, BinaryAlphabet);  double kazakhInformationAmountBinary = EntropyCalculator.CalculateInformationAmount(danishFIOBinary, danishEntropyBinary, BinaryAlphabet);  // Вывод количества информации в ФИО  Console.WriteLine($"3. Количество информации в ФИО:");  Console.WriteLine("----------------------------------------------------");  Console.WriteLine($"Количество информации в ФИО на датском: {danishInformationAmount:F4} бит");  Console.WriteLine($"Количество информации в ФИО на казахском: {kazakhInformationAmount:F4} бит");  Console.WriteLine($"Количество информации в ФИО на датском (бинарный): {danishInformationAmountBinary:F4} бит");  Console.WriteLine($"Количество информации в ФИО на казахском (бинарный): {kazakhInformationAmountBinary:F4} бит");  Console.ResetColor();  Console.WriteLine("----------------------------------------------------\n"); |

Листинг 1.4 – Вычисление количества информации

В итоге получаем:

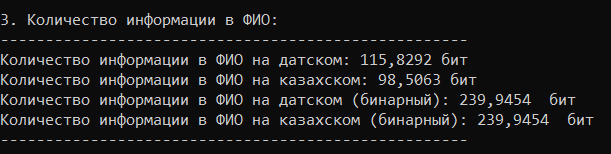
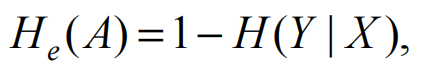


Рисунок 1.7 – Количество информации

г) выполнить задание пункта (в) при условии, что вероятность ошибочной передачи единичного бита сообщения составляет: 0,1; 0,5; 1,0.

Если вероятность ошибки в ДСК отлична от 0 (р > 0), переданное сообщение может содержать ошибки: Хk ≠ Yk. Количество информации в таком сообщении при его передаче по ДСК будет определяться не энтропией двоичного алфавита, а эффективной энтропией алфавита или пропускной способностью канала:



где *H*(*Y* | *X*) *–* условная энтропия:



Запишем метод для поиска эффективной энтропии:

|  |
| --- |
| public static double EffectiveEntropy(double p)  {  if (p == 0 || p == 1)  {  return 0;  }  double q = 1 - p;  return -p \* Math.Log2(p) - q \* Math.Log2(q);  } |

Листинг 1.5 – Вычисление эффективной энтропии

В итоге получаем:

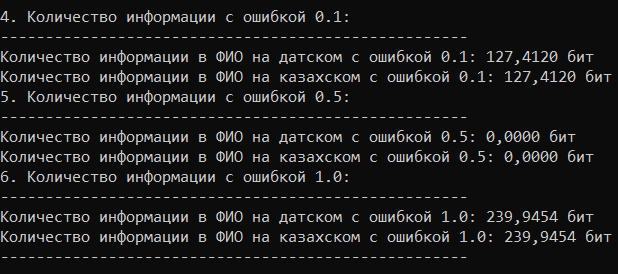


Рисунок 1.7 – Количество информации с ошибками

# Вывод

В ходе выполнения лабораторной работы были закреплены теоретические знания по основам теории информации, а также приобретены практические навыки расчёта ключевых информационных характеристик дискретных систем. Было разработано приложение на языке C#, предназначенное для автоматизации вычислений. С его помощью рассчитывалась энтропия по Шеннону для алфавитов на основе латиницы (итальянский язык) и кириллицы (монгольский язык), а также для их двоичных представлений. Частотный анализ символов исходных текстов был проведён с последующим сохранением результатов в формате Excel.