МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждение образования «БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

**Лабораторная работа № 5**

**ИССЛЕДОВАНИЕ КРИПТОГРАФИЧЕСКИХ**

**ШИФРОВ НА ОСНОВЕ ПЕРЕСТАНОВКИ СИМВОЛОВ**

Разработала: Бай И.О.

ФИТ 3 курс 5 группа

Преподаватель: Савельева М.Г.

Минск 2023

**Цель:** изучение и приобретение практических навыков разработки и использования приложений для реализации перестановочных шифров (работа рассчитана на 4 часа аудиторных занятий).

**Задачи:**

1. Закрепить теоретические знания по алгебраическому описанию, алгоритмам реализации операций зашифрования/расшифрования и оценке криптостойкости перестановочных шифров.
2. Ознакомиться с особенностями реализации и свойствами различных перестановочных шифров на основе готового программного средства (L\_LUX).
3. Разработать приложение для реализации указанных преподавателем методов перестановочного зашифрования/расшифрования.
4. Выполнить исследование криптостойкости шифров на основе статистических данных о частотах появления символов в исходном и зашифрованном сообщениях.
5. Оценить скорость зашифрования/расшифрования реализованных способов шифров.
6. Результаты выполнения лабораторной работы оформить в виде описания разработанного приложения, методики выполнения экспериментов с использованием приложения и результатов эксперимента.

**Практическое задание**

|  |  |  |
| --- | --- | --- |
| Вариант | Алфавит | Шифр |
| 2 | Русский | 1. Маршрутная перестановка (маршрут – спиралью; параметры таблицы – по указанию преподавателя) 2. Множественная перестановка, ключевые слова – собственные имя и фамилия |

***Метод маршрутной перестановки (маршрут – спиралью)***

Организация маршрутной перестановки спиралью представлена на рис. 1.

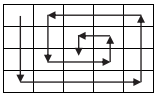


Рис. 1 – Графическое представление метода маршрутной

перестановки спиралью

Описание методов шифрования и расшифрования для данного метода представлены в листинге 1.

function encrypt(matrix) {

  let encryptMess = "";

  let even = true,

    odd = false;

  let timeStart = performance.now();

  ///////////////////////     START     ///////////////////////

  for (let i = 0; i < COLUMS; i++) {

    if (even) {

      for (let j = 0; j < ROWS; j++) {

        encryptMess += matrix[j][i];

      }

      even = false;

      odd = true;

      continue;

    }

    if (odd) {

      for (let j = ROWS - 1; j >= 0; j--) {

        encryptMess += matrix[j][i];

      }

      even = true;

      odd = false;

      continue;

    }

  }

function decrypt(encryptText) {

  let decryptArr = [];

  for (let i = 0; i < ROWS; i++) {

    decryptArr[i] = [];

  }

  let even = true,

    odd = false,

    encryptIndex = 0;

  let timeStart = performance.now();

  ///////////////////////     START     ///////////////////////

  for (let i = 0; i < COLUMS; i++) {

    if (even) {

      for (let j = 0; j < ROWS; j++) {

        decryptArr[j][i] = encryptText[encryptIndex++];

      }

      if (encryptText != encryptText.length) {

        even = false;

        odd = true;

        continue;

      } else {

        break;

      }

    }

    if (odd) {

      for (let j = ROWS - 1; j >= 0; j--) {

        decryptArr[j][i] = encryptText[encryptIndex++];

      }

      if (encryptText != encryptText.length) {

        even = true;

        odd = false;

        continue;

      } else {

        break;

      }

    }

  }

Листинг 1 – Шифрование/расшифрование методом

маршрутной перестановки спиралью

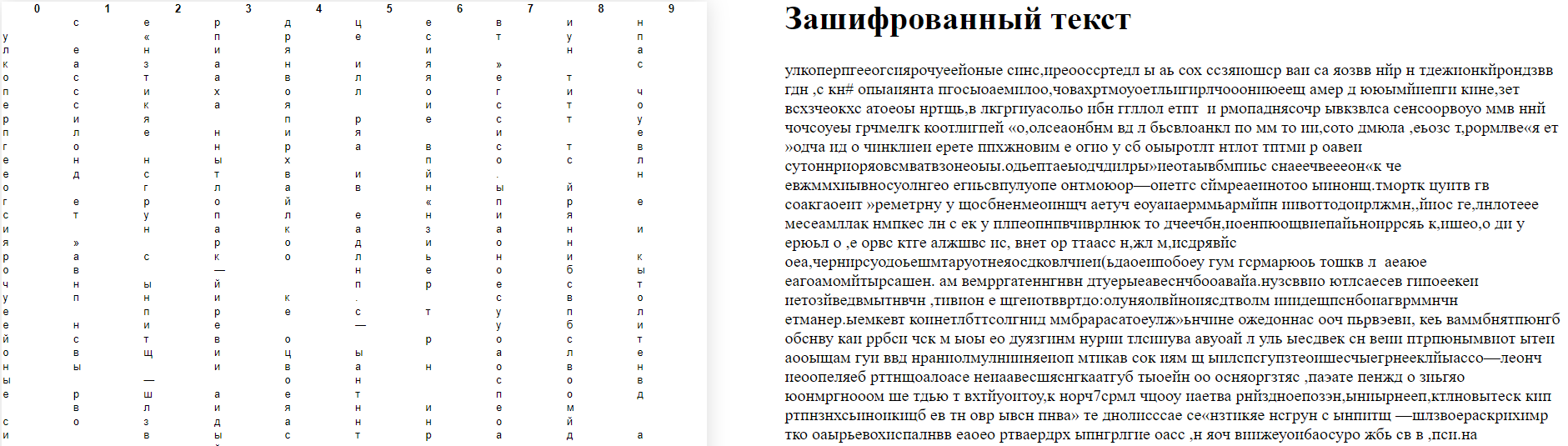


Рис. 2 – Реализация кода программы

**Метод множественной перестановки**

Описание методов шифрования и расшифрования для данного метода представлены в листинге 2.

function multiplyPermutationsEncrypt() {

  let textIndex = 0;

  for (let i = 0; i < ROWS; i++) {

    matrixText[i] = [];

    for (let j = 0; j < COLUMS; j++) {

      if (text[textIndex]) {

        matrixText[i][j] = text[textIndex++];

      }

    }

  }

  let timeStart = performance.now();

  ///////////////////////     START     ///////////////////////

  for (let i = 0; i < ROWS; i++) {

    entable[i] = [];

    for (let j = 0; j < COLUMS; j++) {

      entable[i][j] =

        matrixText[keyNameArr.indexOf(i)][keySurnameArr.indexOf(j)];

    }

  }

  ///////////////////////     FINISH     ///////////////////////

  let timeFinish = performance.now();

  document.querySelector(".encryptTime").innerHTML +=

    timeFinish - timeStart + " мс";

  document.querySelector(".tableInput").innerHTML += printMatrix(matrixText);

  document.querySelector(".tableInput2").innerHTML += printMatrix(entable);

  return entable.flat().join("");

}

function multiplyPermutationsDecrypt(text) {

  let textIndex = 0;

  for (let i = 0; i < ROWS; i++) {

    for (let j = 0; j < COLUMS; j++) {

      if (text[textIndex]) {

        matrixText[i][j] = text[textIndex++];

      }

    }

  }

  let timeStart = performance.now();

  ///////////////////////     START     ///////////////////////

  for (let i = 0; i < ROWS; i++) {

    for (let j = 0; j < COLUMS; j++) {

      entable[i][j] = matrixText[keyNameArr[i]][keySurnameArr[j]];

    }

  }

  ///////////////////////     FINISH     ///////////////////////

  let timeFinish = performance.now();

  document.querySelector(".decryptTime").innerHTML +=

    timeFinish - timeStart + " мс";

  return entable.flat().join("");

}

Листинг 2 – Шифрование/расшифрование методом

множественной перестановки

Реализация кода программы:



Рис. 3 – Реализация кода программы

Графики:

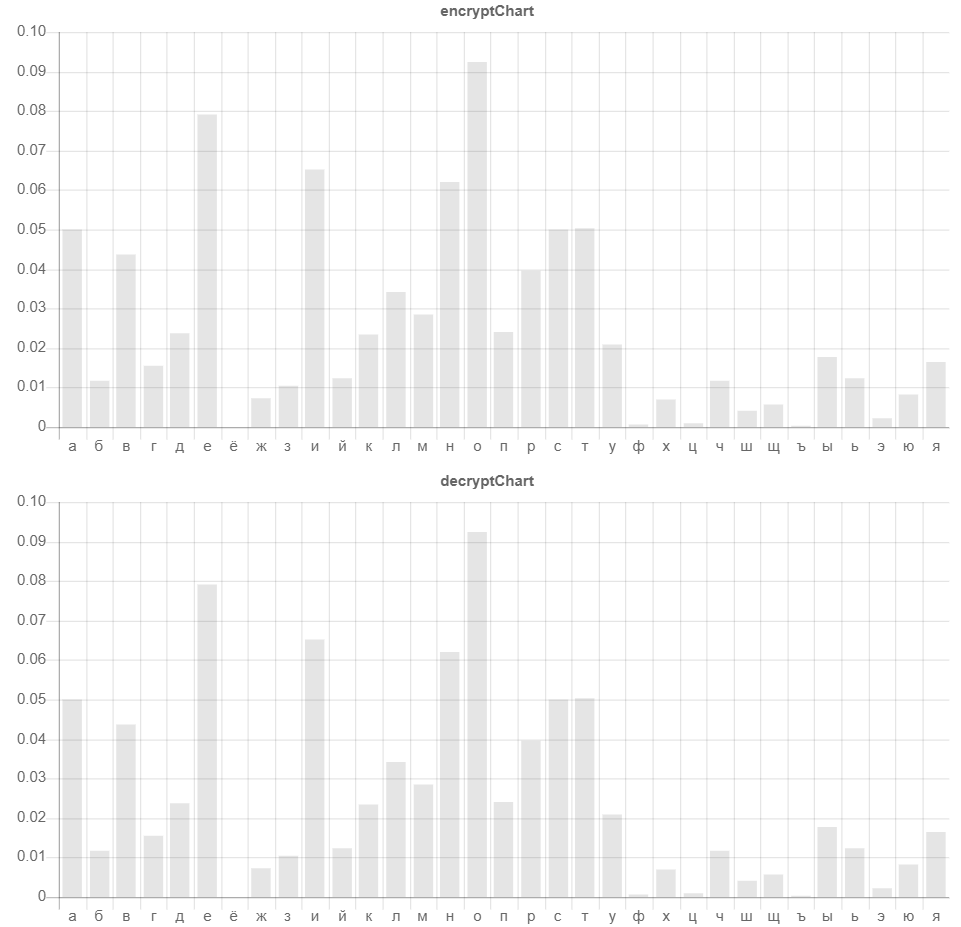


Рис. 4 – Частота появления символов при использовании

маршрутной перестановки спиралью

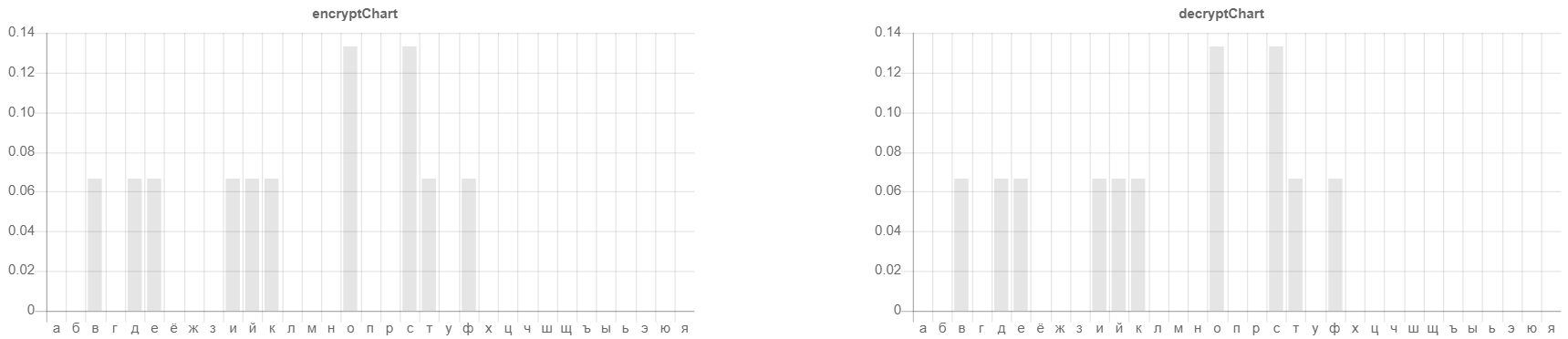


Рис. 5 – Частота появления символов при использовании

множественной перестановки

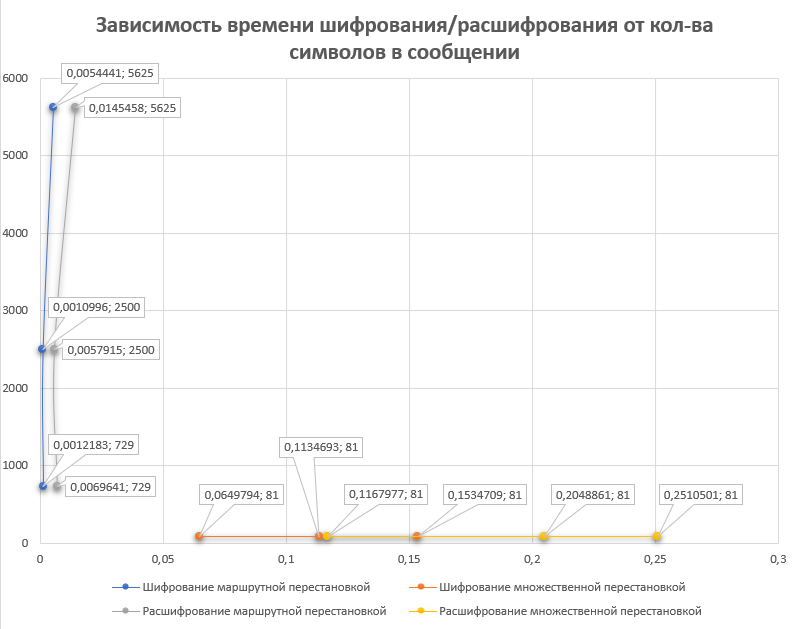


Рис. 6 – Зависимость времени от количества символов в сообщении

**Вывод**: в отличии от шифрования методом подстановки символов, шифрование методом перестановки сохраняет частотность появления символов, т. к. символы входящие в исходное сообщение лишь изменяют свои позиции, в то время как в методах подстановки они заменяются на другие.