

Учреждение образования
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ
УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерного проектирования
Кафедра инженерной психологии и эргономики

КРИПТОГРАФИЧЕСКИЕ ТЕХНОЛОГИИ

Практическая работа №2
Маршрутные и подстановочные шифры

Выполнил:
Глик А. Г.

Проверил:
Давыдович К.И.

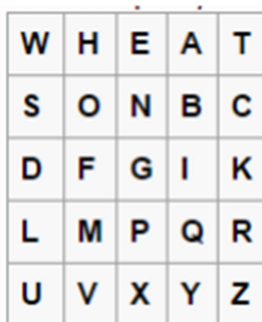
Минск 2023

Цель работы:

Основная часть занятия состоит в закреплении теоретического материала курса, приобретении навыков выполнения задач по основам криптографических технологий, анализа результатов, грамотного оформления отчетов, в частности: изучение алгоритмов и устройства шифров табличной маршрутной перестановки, шифра Плейфера.

Теоретические сведения:

Шифр Плейфера — подстановочный шифр, реализующий замену биграмм. Для шифрования необходим ключ, представляющий собой таблицу. Рассмотрим, в качестве примера следующую таблицу, образующую ключ шифра Плейфера:



W	H	E	A	T
S	O	N	B	C
D	F	G	I	K
L	M	P	Q	R
U	V	X	Y	Z

Рисунок 1 — Ключ-таблица шифра Плейфера

Ход работы:

Задание

1. На основании таблицы, предоставленной на рисунке 1 данного отчета, разработать алгоритм шифрования и дешифрования.
2. Разработать программу, реализующую данный алгоритм на любом языке программирования.

Вывод:

В ходе работы произошло ознакомление с методическим материалом и дополнительной информацией, связанной с предоставленной темой; было разработано приложение, полностью покрывающее задачу, изложенную в вышепредставленных требованиях задания для практического занятия. В приложениях А и Б предоставлены код программы и результат ее работы соответственно.

ПРИЛОЖЕНИЕ А

(обязательное)

Листинг кода программы

```
function findCharIndex(char) {
  for (let i = 0; i < 5; i++) {
    for (let j = 0; j < 5; j++) {
      if (key[i][j] === char) {
        return [i, j];
      }
    }
  }
}

function playfairEncrypt(plainText, key) {

  plainText = plainText.replace(/s/g, "").toUpperCase();

  let pairs = [];
  for (let i = 0; i < plainText.length; i += 2) {
    let firstChar = plainText[i];
    let secondChar = (i + 1 < plainText.length) ? plainText[i + 1] : 'X';
    if (firstChar === secondChar) {
      secondChar = 'X';
      i--;
    }
    pairs.push([firstChar, secondChar]);
  }

  let cipherText = "";
  for (const [char1, char2] of pairs) {
    const [x1, y1] = findCharIndex(char1);
    const [x2, y2] = findCharIndex(char2);

    if (x1 === x2) {
      cipherText += key[x1][(y1 + 1) % 5] + key[x2][(y2 + 1) % 5];
    } else if (y1 === y2) {
      cipherText += key[(x1 + 1) % 5][y1] + key[(x2 + 1) % 5][y2];
    }
  }
}
```

```

    } else {
      cipherText += key[x1][y2] + key[x2][y1];
    }
  }
}

```

```

  return cipherText;
}

```

```

function playfairDecrypt(cipherText, key) {
  let pairs = [];
  for (let i = 0; i < cipherText.length; i += 2) {
    pairs.push([cipherText[i], cipherText[i + 1]]);
  }

```

```

  let plainText = "";
  for (const [char1, char2] of pairs) {
    const [x1, y1] = findCharIndex(char1);
    const [x2, y2] = findCharIndex(char2);

    if (x1 === x2) {
      plainText += key[x1][(y1 + 4) % 5] + key[x2][(y2 + 4) % 5];
    } else if (y1 === y2) {
      plainText += key[(x1 + 4) % 5][y1] + key[(x2 + 4) % 5][y2];
    } else {
      plainText += key[x1][y2] + key[x2][y1];
    } }

```

```

  return plainText;
}

```

```

const key = [['W', 'H', 'E', 'A', 'T'], ['S', 'O', 'N', 'B', 'C'], ['D', 'F', 'G', 'I', 'K'],
['L', 'M', 'P', 'Q', 'R'], ['U', 'V', 'X', 'Y', 'Z']];
const plaintext = "Arseni Glik";
const encryptedText = playfairEncrypt(plaintext, key);
console.log("Зашифрованный текст:", encryptedText);
const decryptedText = playfairDecrypt(encryptedText, key);
console.log("Расшифрованный текст:", decryptedText);

```

ПРИЛОЖЕНИЕ Б
(обязательное)
Результаты работы программы

```
Microsoft Windows [Version 10.0.19045.3448]  
(c) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.  
  
C:\Users\arsen>node D:\code\kgt\PT_2\practical_task2.js  
Зашифрованный текст: TQNWBGDPKD  
Расшифрованный текст: ARSENIGLIK  
  
C:\Users\arsen>
```

Рисунок Б — Результат работы приложения