Лабораторная работа №1

Шифры простой замены

Кубасов В.Ю., ст.б. 1132249516

Содержание

1 Цель работы

Ознакомиться с простейшим и древнейшим вариантом шифрования, как метода защиты передаваемой информации - алфавитными перестановками

2 Задание

Реализовать 2 шифра: - Шифр цезаря - Шифр Атбаш

3 Теоретическое введение

Шифр Цезаря - первый документированный европейский шифр. Является шифром [**простой замены**][1]. Основным назначением шифрования ялвяется защита информации от третьих лиц. Так, например, данный шифр был разработан для безопасной передачи сообщений послами. Используя шифрование исходного письма, гарантировалась недоступность государственной информации, а также однозначное определение смысла послания после получения письма нужным лицом. Аналогичным шифром, относящимся к данному классу является шифр Атбаш, где использовался “перевернутый” алфавит. Пусть сейчас данные шифры являются устаревшими и взлом их осуществляется разными методами, они положили основу криптографии.

Взлом данных шифров осуществляется **частотным анализом**. Частнотный анализ требует значительной подготовки и [большой выборки сообщений][2], совпадающих по языку с зашифрованным сообщением. Противоядием к частнотному анализу может быть банальное несоблюдение норм языка

(нарушение орфографии ведёт к искажению частот появления букв в сообщении, например, бука “о” перестанет встречаться в 45 раз чаще буквы “ф”), либо вставкой лишних символов, не несущих в себе информации. Помимо частотного анализа в настоящее время является возможным использование **bruteforce-метода** - метода грубой силы или полного перебора.

4 Выполнение лабораторной работы

Ввиду того, что шифрование является заменой одного символа на другой, можно выделить несколько вариантов реализации шифра Цезаря и Атбаш: 1. Путём создания хэш-таблицы, где ключ - исходный символ, а значение в паре KeyValuePair - зашифрованным символом. 2. Создание строки str с итоговым шифром, где индекс i - порядковый номре исходного символа в алфавите, а str[i] - зашифрованный символ.

Был реализован 2ой вариант, тогда функция, создающая нужную шифровальную строку со смещением offset выглядит как:

```
function gimmePassword()
    println("Введите смещение");
    offset = parse(Int64, readline());

    trueOffset = mod(offset, smallZOrd - smallAOrd + 1);

    rawPassword = "";

    for i in smallAOrd:1:smallZOrd
        rawPassword = rawPassword * (Char)(i);
    end;

    password = rawPassword[trueOffset + 1:length(rawPassword)] * rawPassword[1:trueOff
```

```
    return password;
end;
```

Аналогично для Атбаш шифра:

```
function gimmePassword()
    rawPassword = "";

    for i in smallAOrd:1:smallZOrd
        rawPassword = rawPassword * (Char)(i);
    end;

    password = reverse(rawPassword * ' ');

    return password;
end;
```

где переменная *password* - результирующая “шифровальная” строка, а константы smallAOrd, bigZOrd - начала и концы алфавита (в прописном и строчном варианте):

```
const smallAOrd = codepoint('a');
const smallZOrd = codepoint('z');

const bigAOrd = codepoint('A');
const bigZOrd = codepoint('Z');
```

Далее полученная в функции строка (в нижнем регистре) копируется в верхний регистр для шифрования строчных и прописных букв:

```
lowerCasePassword = gimmePassword();
upperCasePassword = uppercase(lowerCasePassword);
```

Далее ожидаем на вход строку от пользователя для шифрования (с заданным смещением для шифра Цезаря или без дополнительных входных данных для шифра Атбаш):

```
println("Введите строку для шифрования");
unshieldedString = readline();


shieldedString = "";
```

где *unshieldedStirng* - незашифрованная строка, а *shieldedString* - зашифрованная (введена для задания типа переменной). Далее необходимо посимвольно итерироваться по незашифрованной строке, выбирая по номеру буквы в алфавите соответствующий символ из шифровальной строки:

```
for i in 1:1:length(unshieldedString)
    if (occursin(unshieldedString[i], lowerCasePassword))
        global shieldedString = shieldedString * lowerCasePassword[codepoint(unshielde
    elseif (occursin(unshieldedString[i], upperCasePassword))
        global shieldedString = shieldedString * upperCasePassword[codepoint(unshielde
    else
        global shieldedString = shieldedString * unshieldedString[i];
    end;
end;
```

По итогу двух алгоритмов получаем *shieldedString*, содержащую зашифрованное сообщение, после чего выводим его на экран.

Пример консольного вывода для шифра Цезаря со смещением 5:

```
Введите смещение
5
Введите строку для шифрования
Hi, I'm truly Caesar!
Mn, N'r ywzqd Hfjxfw!
```

5 Выводы

1. Ознакомились с простейшими видами шифрования на прмиере шифра Цезаря и шифра Атбаш.
2. Реализовали данные шифры на языке Julia
3. Выявили слабые и сильные стороны подобных шифров

Список литературы

1. Марков А.С., Цирлов В.Л. Основы криптографии: подготовка к CISSP // Вопросы кибербезопасности. Акционерное общество «Научно-производственное объединение «Эшелон», 2015. № 1 (9). С. 65–73.
2. Авдошин С., Савельева А. Криптоанализ и криптография: история противостояния // Бизнес-информатика. Федеральное государственное автономное образовательное учреждение высшего ..., 2009. № 2. С. 3–11.