

Лабораторная работа 1

Классические криптосистемы

Дедлайн: 08.10.2024

1 Введение

В данной лабораторной работе вам необходимо программно реализовать шесть классических криптосистем:

- [Шифр сдвига](#)
- [Аффинный шифр](#)
- [Шифр простой замены](#)
- [Шифр Хилла](#)
- [Шифр перестановки](#)
- [Шифр Виженера](#)

Данные криптосистемы не отличаются высокой криптостойкостью, однако, хорошо иллюстрируют приемы, которые могут применяться для шифрования сообщений. Дополнительную информацию вы можете получить в материалах лекции и книге «[Основы классической криптологии](#)».

2 Условие лабораторной работы

Ваша задача - реализация всех шести классических криптосистем.

Под реализацией следует понимать компьютерную программу, написанную на одном из доступных языков программирования: {C/C++, C#, Java, Python}, которая обладает определенной функциональностью.

Функциональность программы:

- зашифрование сообщения любой из указанных выше криптосистем;
- расшифрование сообщения, зашифрованного любой из указанных выше криптосистем;
- корректная обработка некорректного алфавита;
- корректная обработка некорректного ключа;
- корректная обработка не входящих в алфавит символов входного текста.

Входные данные программы:

- файл с алфавитом (alphabet.txt);
- файл с открытым текстом / шифртекстом (in.txt);
- файл с ключом (key.txt);
- указание, какую нужно проводить операцию (зашифрование или расшифрование);
- указание, какую криптосистему нужно использовать.

Выходные данные программы

- файл с открытым текстом (decrypt.txt) / шифртекстом (crypt.txt) в зависимости от выбранной операции;
- либо сообщение об успешном выполнении операции, либо сообщение об ошибке (с указанием причины ошибки).

3 Описание входных данных

3.1 Файл с алфавитом

Файл с алфавитом содержит единственную строку, в которой записан набор попарно различных символов. Это могут быть заглавные и строчные символы латинского и русского алфавитов, а также спецсимволы. По умолчанию (нет файла alphabet.txt) должен использоваться следующий алфавит:

33 заглавные русские буквы от А до Я, а также символ ' ' (пробел):

«АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ _».

При этом считаем, что 'А' = 0, 'Б' = 1, ..., 'Я' = 32, ' ' = 33.

Следует иметь в виду, что файл с алфавитом может иметь некорректный формат.

Программа должна каким-либо образом корректно обработать эту ситуацию. Далее будем считать, что алфавит имеет мощность M.

3.2 Файл с открытым текстом / шифртекстом

Файл содержит единственную строку, в которой записан открытый текст / шифртекст, состоящий из символов алфавита. При этом следует учитывать, что открытый текст / шифртекст может иметь некорректный формат.

Программа должна каким-либо образом корректно обработать эту ситуацию.

3.3 Файл с ключом

Файл содержит единственную строку, в которой записан ключ. Для каждой криптосистемы ключ имеет собственную форму записи. Ниже описаны форматы корректных ключевых данных для каждой из криптосистем. При этом следует учитывать, что ключ может иметь некорректный формат.

Программа должна каким-либо образом корректно обработать эту ситуацию.

3.3.1 Шифр сдвига

В качестве ключа задается символ из алфавита. При этом его числовое представление: $k \in \mathbb{Z}_M$.

3.3.2 Аффинный шифр

В качестве ключа задается пара символов из алфавита (k_1, k_2) , записанных подряд (без разделителей). При этом их числовое представление $(k_1, M) = 1; k_1, k_2 \in \mathbb{Z}_M$.

3.3.3 Шифр простой замены

В качестве ключа задается вторая строка подстановки. При этом она должна содержать все символы алфавита, записанные в произвольном порядке по одному разу каждый, а числовое представление любого символа ключа: $k_i \in \mathbb{Z}_M$.

3.3.4 Шифр Хилла

В качестве ключа задаются четыре символа из алфавита, $(c_1, c_2, c_3, c_4) \sim (k_{11}, k_{12}, k_{21}, k_{22})$, записанных подряд (без разделителей). Из них и формируется ключевая матрица

$$K = \begin{pmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{pmatrix}$$

При этом матрица K обратима, т.е. $\exists \det(K)^{-1} \Leftrightarrow \det(K) \in \mathbb{F}_M^*$.

Если длина открытого текста не кратна 2, то в его конец необходимо дописать любой символ алфавита.

Внимание! Для шифрования следует умножать вектор с открытым текстом на матрицу-ключ (но не наоборот!), а для расшифрования - вектор с шифротекстом на обратную матрицу ключа (но не наоборот!).

3.3.5 Шифр перестановки

В качестве ключа задается строка, состоящая из попарно различных символов алфавита, длиной не более M символов. Положение символов ключа в алфавите определяет нижнюю строку подстановки. При этом числовое представление любого символа ключа $k_i \in \mathbb{Z}_M$. Если длина открытого текста не кратна длине ключа, то в его конец необходимо дописать один и тот же символ алфавита таким образом, чтобы длина получившегося открытого текста была кратна длине ключа.

3.3.6 Шифр Виженера

В качестве ключа задается строка произвольной длины, состоящая из символов алфавита. При этом числовое представление любого символа ключа $k_i \in \mathbb{Z}_M$.

3.4 Указания

Указания, определяющие криптосистему и процедуру (зашифрование или расшифрование) могут поступать на вход любым удобным автору программы способом, в том числе из консоли, из файла и через оконный интерфейс.

4 Бонусные задания

4.1 Бонусное задание №1

Реализовать программу, которая применяет частотный анализ, для раскрытия (взлома) шифртекстов, полученный с помощью шифра простой замены. Необходимо получить программу, на вход в которую поступает только шифртекст, зашифрованный шифром простой замены, а на выходе программа возвращает

открытый текст и ключ. Язык открытых текстов: английский или русский (по выбору выполняющего задание).

4.2 Бонусное задание №2

Реализовать программу, которая применяет метод Касиски, для раскрытия (взлома) шифртекстов, полученный с помощью шифра Виженера. Необходимо получить программу, на вход в которую поступает только шифртекст, зашифрованный шифром Виженера, а на выходе программа возвращает открытый текст и ключ. Язык открытых текстов: русский.

4.3 Бонусное задание №3

Реализовать программу, которая применяет метод индекса совпадений, для раскрытия (взлома) шифртекстов, полученный с помощью шифра Виженера. Необходимо получить программу, на вход в которую поступает только шифртекст, зашифрованный шифром Виженера, а на выходе программа возвращает открытый текст и ключ. Язык открытых текстов: русский.

4.4 Бонусное задание №4

Реализовать программу, которая используется для раскрытия (взлома) шифртекстов, полученный с помощью шифра Хилла. Необходимо получить программу, на вход в которую поступает только шифртекст, зашифрованный шифром Хилла, а на выходе программа возвращает открытый текст и ключ. Известно, что для шифрования всегда используется матрица 2×2 . Язык открытых текстов: русский.

4.5 Особенности бонусных заданий

- Бонусное задание №1 может выполнить неограниченное число студентов.
 - Бонусное задание №1 можно выполнять как индивидуально, так и парами (в этом случае каждый получит по половине баллов).
 - За бонусное задание №1 можно получить 15-25 баллов в зависимости от точности расшифрования.
 - За бонусные задания №2–4 можно получить по 20 баллов.
 - Бонусные задания №2–4 могут выполнить по два студента, которое первыми напишут о своем желании в группе Telegram.
- При этом два студента, которые делают одно и то же задание, должны использовать **!разные!** языки программирования.
- Один студент может делать не более одного задания (из бонусных заданий №2–4).
 - Бонусные задания можно выполнить и сдать до 25.11.24 включительно.

5 Порядок сдачи лабораторной работы

Работа сдается лично преподавателю. Исходный код (каждой отдельной) программы должен содержаться в едином исходном файле с названием *lab<n>_<фамилия>_<назначение>*. (Пример: *lab2_Ivanov_MainPart.cpp* или *lab3_Sidorov_BonusTask4.py*). Компиляция (интерпретация) программы должна выполняться из явно указанного исходного файла без ошибок. При наличии любых частей кода выполненных не самостоятельно (в т.ч. если код написан самостоятельно, но по его структуре консультировались с другими студентами), преподаватель должен быть уведомлен заблаговременно, и соответствующие комментарии **!обязаны!** быть указаны в исходном коде.

- **Обычный порядок сдачи:** в качестве срока сдачи используется дата в которую вы сдаете лабораторную работу преподавателю.
 - **Досрочный порядок сдачи:** при помощи программы, предоставленной преподавателем (HashCalc), для исходного файла вычисляется контрольная характеристика. В группу Telegram отправляется сообщение в формате *#TaskDone <имя исходного файла с расширением> <значение контрольной характеристики>*. (Пример: *#TaskDone lab2_Ivanov_MainPart.cpp K3aSsaHCTjC4q8r/X/ZhZlkMxqGYZtzhMjHHeq5c/o=*)
- Работа сдается очно, при соблюдении всех условий (в том числе совпадении значения контрольной характеристики), в качестве даты сдачи используется дата сообщения в Telegram. Сообщения с пометкой “редактировано” не принимаются!