Модуль 4. Алгоритмы и структуры данных

Тема4.12.* Введение в криптографию и криптоанализ

Оглавление

4.12. Введение в криптографию и криптоанализ 2
4.12.1. Шифры подстановки 2
Шифр Цезаря 2
ROT13 3
4.12.2. Шифры перестановки 4
Решетка Кардано 4
Конь Эйлера 4
Шифр Сцитала 4
4.12.3. Шифрование и ЭВМ 5
4.12.4. Криптоанализ 6
Таблицы частотности для русского и английского языков 7
Пример криптоанализа.. 7
Упражнение 4.12.1 8
Упражнение 4.12.2 10
Упражнение 4.12.3 12
Задание 4.12.1 13
Благодарности 13

4.12. Введение в криптографию и криптоанализ

Шифрование — это обратимое преобразование текста (информации) с целью сокрытия информации от посторонних.

Наивные шифры чаще всего основаны на том, что постороннее лицо не знает метода шифрования.

Современное шифрование основано на трудности расшифровки даже при опубликованном методе шифрования.

На протяжении всего процесса развития цивилизации интерес к сокрытию информации всегда был очень высок. Вариантов было очень много. Например разного рода тайнопись, то есть способ сделать текст невидимым для постороннего - например писать секретные сведения молоком на бумаге между строк; или татуировка на обритой голове раба (впоследствии выросшие волосы маскировали сообщение). Однако гораздо эффективнее были шифры, так как раскрытие информации было практически невозможным без алгоритма и/или ключа к шифру.

4.12.1. Шифры подстановки

Простой подстановочный шифр (или простой замены, моноалфавитный шифр) — класс методов шифрования, которые сводятся к созданию по определённому алгоритму таблицы шифрования, в которой для каждой буквы открытого текста существует единственная сопоставленная ей буква шифр-текста. Само шифрование заключается в замене букв согласно таблице. Для расшифровки достаточно иметь ту же таблицу, либо знать алгоритм, по которой она генерируется.

К шифрам простой замены относятся многие способы шифрования, возникшие в древности или средневековье, как, например, Атбаш или Шифр Цезаря. Для вскрытия подобных шифров используется частотный криптоанализ.

Шифр Цезаря[1]

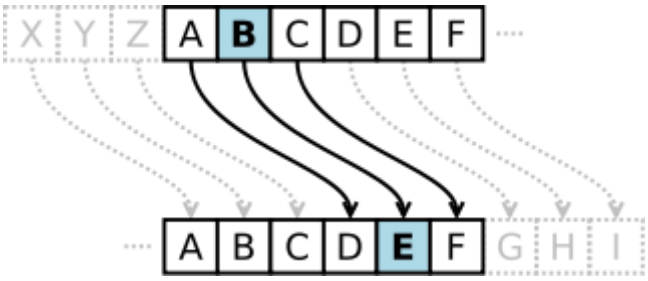
Шифр Цезаря — один из самых простых и наиболее широко известных методов шифрования. Активно использовался императором Гаем Юлием Цезарем в переписке со своими военачальниками в древнем Риме (50г до н.э.).

Шифр Цезаря — это вид шифра подстановки, в котором каждый символ в открытом тексте заменяется символом, находящимся на некотором постоянном числе позиций левее или правее него в алфавите.

Это постоянное число позиций сдвига при шифровании называется **ключом** шифра Цезаря.

Например, для русского алфавита в шифре со сдвигом вправо с ключом 3 (именно такой ключ применял Цезарь), А будет заменена на Г, Б станет Д, и так далее.

[1]Шифр Цезаря [Электронный ресурс] : Материал из Википедии — свободной энциклопедии : Дата обновления: 15.01.2015 // Википедия, свободная энциклопедия. — Электрон. дан. — Сан-Франциско: Фонд Викимедиа, 2015. — Режим доступа: http://ru.wikipedia.org/?oldid=67953711



Шифр Цезаря со сдвигом на 3:

A заменяется на D
B заменяется на E
...
Z заменяется на C

Итак, подведем итог. Получается, что для дешифрования сообщения, закодированного шифром Цезаря, нужно знать примененный:

- алфавит;
- ключ;
- направление сдвига (вправо или влево).

ROT13

Современная разновидность шифра Цезаря - ROT13 (от англ. «*rotate*»). Число 13 в наименовании шифра означает, что ключ равен 13.

Этот шифр подстановкой широко используется в интернет-форумах, как средство для сокрытия спойлеров, основных мыслей, решений загадок от случайного взгляда. ROT13 был охарактеризован как «сетевой эквивалент того, как в журналах печатают ответы на вопросы викторин — перевёрнутыми буквами».

При алгоритме ROT13 каждый буквенный символ английского алфавита заменяется на соответствующий ему со сдвигом на 13 позиций. А цифры, пробелы и все остальные символы остаются без изменений.

Почему в качестве ключа выбрано 13? Потому что он разрабатывался для английского алфавита, а там всего 26 букв, а 26 = 2 × 13. Такой ключ позволяет получить так называемый “**взаимный шифр**”- когда исходное и закодированное сообщение можно получить друг из друга, применив один и тот же алгоритм.

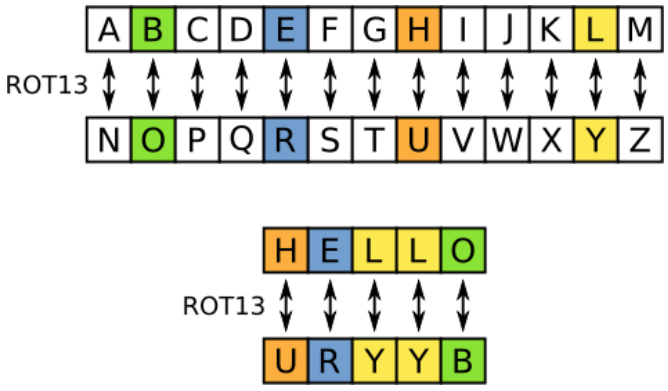
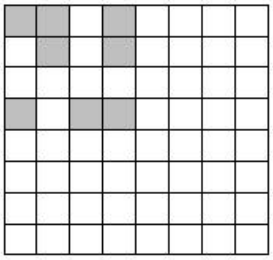


Рисунок. ROT13 - шифровка и расшифровка происходит одинаково

4.12.2. Шифры перестановки

В качестве альтернативы шифрам подстановки можно рассматривать перестановочные шифры. В них, элементы текста переставляются в ином от исходного порядке, а сами элементы остаются неизменными. Тогда как в шифрах подстановки, элементы текста не меняют свою последовательность, а изменяются сами.

Решетка Кардано



Этот способ шифрования был придуман итальянским математиком Джероламо Кардано. Он представлял собой трафарет с прорезанными окошками, через которые на лист бумаги последовательно записывался текст. После заполнения всех окошек трафарет поворачивали на 90^0. И так три раза, после чего решетку Кардано убираем и клетки, оставшиеся пустыми, заполняем "мусором": различными буквами, знаками препинания, цифрами (в произвольном порядке, чем хаотичнее, тем лучше). Например зашифруем сообщение «шифрование с помощью решетки Кардано»:

ш	и		ф				
	р		о				
в		а	н				

ш	и		ф	и			е
	р		о			с	п
				о			
в		а	н	м		о	щ

ш	и		ф	и			е
	р		о			с	п
				о			
в		а	н	м		о	щ
и	к		а	ь	ю		р
			р				
д	а			е		ш	
н			о	е		т	к

ш	и	к	ф	и	т	с	е
м	р	в	о	п	а	с	п
в	и	р	а	о	ц	ф	ю
в	я	а	н	м	п	о	щ
и	к	ц	а	ь	ю	э	р
и	с	х	р	г	л	е	х
д	а	ю	ж	е	ы	ш	з
н	ч	й	о	е	й	т	к

Конь Эйлера

Задача, на основе которой и составлен шифр, заключается в нахождении маршрута шахматного коня, проходящего через все поля доски только один раз. Этот маршрут и является порядком заполнения текстом квадратной матрицы 8x8. Затем, выписывается текст слева-направо. Максимальная длинна сообщения — 64 символа.

Шифр Сцитала

Очень удачным примером шифра перестановки является шифр Сцитала, использовавшийся еще во времена Древней Спарты. Ключом такого шифра была цилиндрическая палочка, а шифрование выполнялось следующим образом:

- узкая пергаментная лента наматывалась по спирали на цилиндрическую палочку;
- шифруемый текст писался на пергаментной ленте по длине палочки, после того как длина палочки оказывалась

исчерпанной, она поворачивалась и текст писался далее, пока либо не заканчивался текст, либо не исписывалась вся пергаментная лента. В последнем случае использовался очередной кусок пергаментной ленты.

Расшифровка выполнялась с использованием палочки такого же диаметра. Таким образом, длина блока n определялась длиной и диаметром палочки, а само шифрование заключалось в перестановке символов исходного текста в соответствии с длиной окружности палочки.

Например, используя палочку, по длине окружности которой помещается 4 символа, а длина палочки позволяет записать 6 символов, исходный текст: «это шифр древней спарты» превратится в шифрограмму: «эфвптрнао ер дйтшр ыиес». Длина блока n = 23, а вектор t, указывающий правило перестановки, для этого шифра может быть записан следующим образом: t = {1, 7, 13, 19, 2, 8, 14, 20, 3, 9, 15, 21, 4, 10, 16, 22, 5, 11, 17, 23, 6, 12, 18}.

4.12.3. Шифрование и ЭВМ

История использования вычислительных машин восходит к времени Второй мировой войны. В это время Германия использовала передовую по тем временам электрическую шифровальную машину «Энигма» для обмена сообщениями в войсках. На тот момент без учёта настройки положения колец (нем. Ringstellung), количество различных ключей составляло 10^{16}.

В течении 1920-1930 г.г. группа польских математиков искала способ дешифровки сообщений и достигла некоторых успехов. В частности они догадались использовать машины подобные Энигме для расшифровки сообщений. Чуть позже Алан Тьюринг создал дешифровальную машину Bombe на основе этого прототипа.

Дальнейшая работа по взлому была организована в главном шифровальном подразделении Великобритании — Правительственной школе кодов и шифров Блетчли-парк (центр «Station X»). В разгар деятельности Блетчли-парк насчитывал 12 тысяч человек, но, несмотря на это, немцы не узнали о нём до самого конца войны! Сообщения, расшифрованные центром, имели гриф секретности «Ultra» — выше, чем использовавшийся до этого «Top Secret». Англичане предпринимали повышенные меры безопасности, чтобы Германия не догадалась о раскрытии шифра.

Начиная с этого времени вся работа по шифрованию/расшифровыванию и криптоанализу ведется только с применением ЭВМ. Конечно изменились алгоритмы, подходы, матаппарат, но не изменилось роль шифрования в жизни стран и людей.

	*Ярким эпизодом деятельности Блетчли-парк является случай с бомбардировкой Ковентри 14 ноября 1940 года, о которой премьер-министру Великобритании Уинстону Черчиллю было известно заранее благодаря расшифровке приказа. Однако Черчилль, опираясь на мнение аналитиков о возможности Германии догадаться об операции «Ультра», принял решение о непринятии мер к защите города и эвакуации жителей.* *“Война заставляет нас все больше и больше играть в Бога. Не знаю, как бы я поступил…”* *Президент США Франклин Рузвельт о бомбардировке Ковентри*

	Для СССР существование и даже результаты работы «Station X» секрета не представляли. Именно из результатов сообщений, дешифрованных в «Station X», СССР узнал о намечающемся «реванше» Гитлера за Сталинградскую битву и смог подготовиться к операции на Курском направлении, получившем названии «Курская дуга». *Посмотреть на работу Энигмы можно здесь* http://enigmaco.de/enigma/enigma.html.

4.12.4. Криптоанализ

Криптоанализ - наука о методах расшифровки зашифрованной информации без предназначенного для такой расшифровки ключа. Первоначально методы криптоанализа основывались на лингвистических закономерностях естественного текста и реализовывались с использованием только карандаша и бумаги. Со временем в криптоанализе нарастает роль чисто математических методов, для реализации которых используются специализированные криптоаналитические компьютеры.

Попытку раскрытия конкретного шифра с применением методов криптоанализа называют ***криптографической атакой*** на этот шифр. Криптографическую атаку, в ходе которой раскрыть шифр удалось, называют ***взломом*** или ***вскрытием***.

Хотя понятие криптоанализ было введено сравнительно недавно, некоторые методы взлома были изобретены десятки веков назад. Первым известным письменным упоминанием о криптоанализе является «Манускрипт о дешифровке криптографических сообщений», написанный арабским учёным Ал-Кинди ещё в 9 веке. В этом научном труде содержится описание метода частотного анализа.

Частотный анализ — основной метод атаки на большинство классических шифров перестановки или замены. Этот метод основывается на предположении о том, что каждый символ алфавита встречается в тексте неодинаковое количество раз и в открытом тексте, и в шифротексте. При этом при условии достаточно большой длины шифрованного сообщения шифры, основанные на одном алфавите, легко поддаются частотному анализу: если частота появления буквы в языке (смотри ниже таблицы частотности) и частота появления некоторого присутствующего в шифротексте символа приблизительно равны, то в этом случае с большой долей вероятности можно предположить, что данный символ и будет этой самой буквой.

Применение высокопроизводительных вычислительных систем сделало возможным вскрытие слабых шифров ***методом простого перебора*** (*brute force*), так например был взломан шифр CSS— системы защиты цифрового медиаконтента на DVD-носителях.

Совершенствование математического аппарата криптоанализа позволило например создать программное обеспечение, которое даже на не особо производительных вычислительных системах, доступных частным лицам, может атаковать протоколы шифрования WiFi WEP/WPA2PSK. Например персональный компьютер с 4мя видеокартами AMD HD7770 способен перебирать несколько миллиардов паролей WPA в час при помощи алгоритма радужных таблиц (rainbow tables) ПО Pyrit.

Как следствие большинство современных алгоритмов шифрования (AES,3DES,Blowfish,ГОСТ 28147—89) имеют достаточно высокую криптостойкость (порядка 2^{224} вариантов ключа), для того чтобы их взлом стал слишком длительным (годы-сотни лет) и слишком дорогим (потребуется суперкомпьютер).

Кроме криптоанализа есть и иные способы атак на шифры и протоколы. Это например похищение ключа человеком или вирусом или нахождение дефекта в алгоритме, который может резко снизить сложность взлома.

Также государствами законодательно регулируется использование средств шифрования, что дает возможность спецслужбам контролировать передачу информации частных лиц. Например система **Carnivore** в США или **СОРМ-2** в России, установленная у всех Интернет-провайдеров и операторов связи, позволяет отслеживать весь интернет трафик. Кроме того в Росии Федеральный закон №40 «О

Федеральной службе безопасности» в статье 13 ограничивает производство и ввоз оборудования с сильными алгоритмами шифрования.

Таблицы частотности для русского и английского языков

Русский язык	
буква	частотность
о	0,10983
е	0,08483
а	0,07998
и	0,07367
н	0,067
т	0,06318
с	0,05473
р	0,04746
в	0,04533
л	0,04343
к	0,03486
м	0,03203
д	0,02977
п	0,02804
у	0,02615
я	0,02001
ы	0,01898
ь	0,01735
г	0,01687
з	0,01641
б	0,01592
ч	0,0145
й	0,01208
х	0,00966
ж	0,0094
ш	0,00718
ю	0,00639
ц	0,00486
щ	0,00361
э	0,00331
ф	0,00267
ъ	0,00037
ё	0,00013

English	
Letter	frequency
a	0,08167
b	0,01492
c	0,02782
d	0,04253
e	0,12702
f	0,02228
g	0,02015
h	0,06094
i	0,06966
j	0,00153
k	0,00772
l	0,04025
m	0,02406
n	0,06749
o	0,07507
p	0,01929
q	0,00095
r	0,05987
s	0,06327
t	0,09056
u	0,02758
v	0,00978
w	0,0236
x	0,0015
y	0,01974
z	0,00074

Пример криптоанализа..

Расшифровка, вариант I

Возьмём какое-либо email сообщение и зашифруем его шифром Цезаря. Это можно сделать например при помощи онлайн утилиты шифрования http://planetcalc.ru/1434/ROT3. Например зашифрованный текст:

```
Жсдуюм жзря. Ргтсплргб, ъхс кгехуг 10ёс врегув ц ргф жстсорлхзоярсз кгрвхлз тс ССТ л ъцхя-ъцхя тс тусзнхгп. Нхс псйзх тулшсжлхз н 14:00 Ф цегйзрлзп Ж.Е.Вщзрнс
```

Получив такое сообщение попробуем проанализировать и получить исходное сообщение. Для начала сделаем несколько допущений.

- Язык оригинала — русский.
- Поскольку явно прослеживается структура текста — сообщение скорее всего не подвергалось сжатию или другой обработке.
- Судя по внешнему виду мы скорее всего имеем дело с алгоритмом простой подстановки.

Исходя из вышесказанных допущений и зная что перед нами письмо электронной почты (а такого рода информацию далеко не всегда можно получить), можно предположить, что письмо начинается с приветствия. Какие два слова (6 и 4е буквы) могут означать приветствие? Наверняка *Жсдуюм жзря. == Добрый день*. Итак, мы получили первые 9 пар подстановки:

Ж	с	д	у	ю	м	з	р	я
Д	о	б	р	ы	й	е	н	ь

Обратите внимание, что все подстановочные буквы имеют сдвиг от оригинала на 3 позиции в алфавите . То есть скорее всего это сдвиговый шифр Цезаря со сдвигом 3. Заменим все буквы в исходном сообщении на буквы со сдвигом назад на 3 и получим оригинал сообщения:

```
Добрый день. Напоминаю, что завтра 10го января у нас дополнительное занятие
по ООП и чуть-чуть по проектам. Кто может приходите к 14:00 С уважением
Д.В.Яценко
```

Обратите внимание что в первом варианте расшифровки мы знали что перед нами письмо. И еще нам очень повезло, что в исходном письме остались пробелы, знаки препинания и большие буквы. А вот если бы ничего этого не было?

Упражнение 4.12.1

Зашифруйте шифром Цезаря какое либо сообщение. Проведите криптоанализ

зашифрованного сообщения. Пример зашифрованного сообщения:

```
жйзмпфдсржеютсорвзхкгжгрлзефхгпдцозфтгфгвйифхнлмжлфнфсфтлфнспгёзрхседулх
грфнсмугкезжнлтсжтулнуюхлзпесеузпвфшегхнлфргипрлнсптгхулфсплепгрлтзррлргтгу
рлщгдсржгтстулнгкцпфхузовзхергипрлнгрстстгжгзхедсржггёзрхтгжгзхфпсфхгескзусеп
лбзёсфълхгбхтсёлдылпргипрлнцфнсоякгзхфжлфнспъзузкрзнсхсузеузпвеплбцкргбхсд
цхзънзлрчсупгщллшгнзуюеюногжюегбхлпзргелрхзурзхсдзьгвфссдьгхянгйжцбрзжзоб5р
сеюшлпзресеузпвефхузълплёгузхгпаоосулёогеюсдэзжлриррсёсугкезжюегхзоярсёснсп
лхзхгёгузхфхгелхтсжфспрзрлззинсптзхзрхрсфхялтузжогёгзхтсъихрцбсхфхгенцпсхнгк
юегзхфвжгегвтсрвхяъхсжсойргтзузжцшсжсптулезфхлжзогетсувжснфугкцтсфозефхузъ
леыхгднегухлузплбсуёгрлксегрекуюетулезжылмнпрсёсълфозррюпйзухегпфузжлфсхуц
жрлнсеугкезжнлерзкгтрсегрёолбескеугьгзхфвеюйлеылмдсржгрголкфлхцгщллтснгкюег
зхъхсезусвхрззефзёскгхзуусулфхлъзфнлплгнхгплфхслхдюеылмфсхуцжрлнплбдсржкгр
сесфжгиханкгпзрргжстцфннугкезжюегхзоярсмугдсхзрзфпсхувргхсъхсжйзмпфрзтусшс
жлханкгпзрпргтугеовзхзёсргрсесзкгжгрлзгёзрхкргнсплхфвфрсеюпнлтсоцъгзхфргувйзр
лзфтзщлгоярюмтлфхсозхлугжлстзозрёгхсудсржсхтугеовзхфвеыгршгмёжзеюфозйлегзхт
гхулфглцдлегзхзёссхтгхулфгфозжезжихннлдзухзуусулфхцхявёсусжулёзфцдюеызпцгёз
рхцплблжгерзпцкргнспспцптсфозцфтзырсмстзугщллепгнгслргсфхусезебйрснлхгмфнсп
псузфлояеццжгихфвкгжзуйгхялжсфхгелхяеосржсрешсжзкгжзуйгрлвфлояегргпзнгзхдс
ржцсхспъхспрздзузйихфеслшгёзрхселзёсфгпсёстсжфхгелолугжлеютсорзрлвегйрсмстзу
гщллгфздвлдсржгфугерлегзхфнуюфсдсвплргжстусфзфлояегугффнгкюегзхъхсдцжцъле
тозрцтюхгофвтснсръхяффсдсмтуснцфленгтфцоцфщлгрлжспрсвжолыяфтусесщлусегос
```

фхзсплзолхезушрзмъзобфхллозесмфнцосесмнсфхлеузкцояхгхзъзёссреюрцйжзррсфлхя
ефхгерцбъзобфхяфпзхгоолъзфнсмфнцосмргрзмефнсузтузфхцтрлнццжгихфвдзйгхятув
пслкыхгднегухлуюплблцмхлъзузктсжкзпрюмшсжлолрллпзхустузфозжцвзёсдсржтсрлпг
зхфлояегтсфхсвррсстзузйгзхзёсдогёсжгувхспцъхснсрхусолуцзхнсптябхзурцбфзхядулх
грфнсмугкезжнлефиетосхяжстозрзрлваозпзрхютогргфлояегтюхгзхфвсфцьзфхелхятснц
ызрлзргйлкряпесеузпвкгфзжгрлвтгуогпзрхфнсёснсплхзхгдсржетзузфхузонзфтгфгзхпл
цесклхзиеусжсесзтспзфхяздсржсееысхогржллсфсдрвнфнгмчсоофлояезргпзузррссфхге
овбхфозженсптябхзурсмфзхлпл6

Тут расшифровать как в первом варианте угадыванием слов не получиться. Что же делать? Если мы догадываемся что шифр сдвиговый, то можно попробовать по очереди расшифровать заменяя буквы со сдвигом, на 1, потом на 2, потом на 3 и т. д. Это режим грубой силы (brute force). Однако он применим если шифр сдвиговый, а если он просто с фиксированными парами (по случайно выбранной таблице)?

В этом случае придется применить в частотный анализ. Для этого посчитаем частоту появления каждого символа в сообщении и поместим в таблицу. По формуле:

Частотность=(кол-во вхождений буквы)/(длинна сообщения)

с	0,0962	п	0,0378	ё	0,0147	ш	0,0049
з	0,0908	ж	0,0358	я	0,0142	щ	0,0039
г	0,0839	т	0,0334	ю	0,0137	ь	0,0020
л	0,0731	н	0,0329	м	0,0123	а	0,0020
х	0,0726	о	0,0319	ъ	0,0108	ч	0,0010
р	0,0653	ц	0,0236	й	0,0088	э	0,0005
ф	0,0579	в	0,0186	б	0,0074		
е	0,0545	к	0,0177	и	0,0069		
у	0,0486	д	0,0167	ы	0,0059		

Сравним полученную таблицу частотности с таблицей частотности русского языка. Особенно в начале таблиц заметно соответствие:

с->о з ->е г->а л->и

То есть как раз сдвиг на три позиции в алфавите. Пробуем расшифровать текст шифром ROT3 и получаем

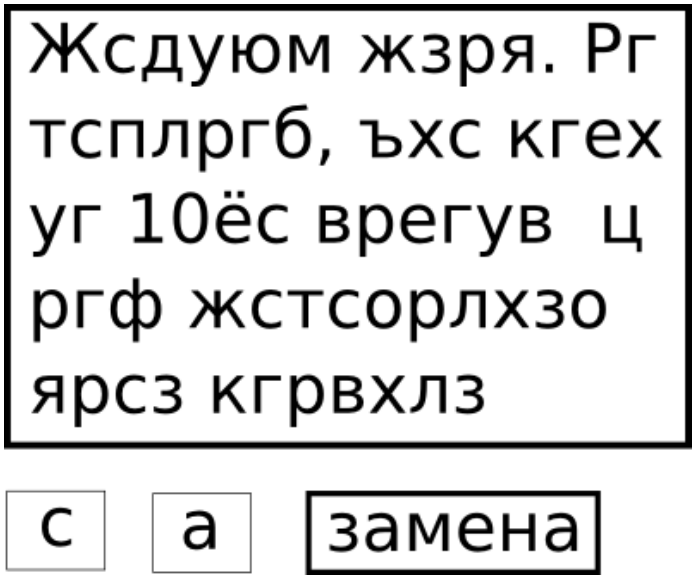
джеймсбондвыполняетзаданиевстамбулеспасаяжёсткийдисксоспискомагентовбританс
койразведкиподприкрытиемвовремясхваткиснаёмникомпатрисомивманипеннинапарни
цабондапоприказумстреляетвнаёмниканопопадаетвбондаагентпадаетсмоставозеровми
бегосчитаютпогибшимнаёмникускользаетсдискомчерезнекотороевремявмибузнаютобу
течкеинформациихакерывыкладываютименавинтернетобещаясообщатькаждуюнеделю
5новыхименвовремявстречимигаретамэллориглавыобъединённогоразведывательногок
омитетагаретставитподсомнениеёкомпетентностьипредлагаетпочётнуюотставкумотка
зываетсядаваяпонятьчтодолжнапередуходомпривестиделавпорядоксразупослевстречи
вштабквартиремиборганизованвзрывприведшийкмногочисленнымжертвамсредисотруд
никовразведкивнезапнованглиювозвращаетсявыжившийбонданализситуациипоказыв
аетчтовероятнеевсегозатеррористическимиактамистоитбывшийсотрудникми6бондзано
восдаётэкзаменнадопусккразведывательнойработенесмотрянаточтоджеймснепроходит
экзаменмнаправляетегонановоезаданиеагентзнакомитсясновымкиполучаетснаряжени
еспециальныйпистолетирадиопеленгаторбондотправляетсявшанхайгдевыслеживаетпа
трисаиубиваетегоотпатрисаследведёткибертеррористутьягородригесубывшемуагенту
мибидавнемузнакомомумпослеуспешнойоперациивмакаоинаостровевюжнокитайскомм
оресильвуудаётсязадержатьидоставитьвлондонвходезадержаниясильванамекаетбонду
отомчтомнебережётсвоихагентовиегосамогоподставилирадивыполненияважнойоперац

ииасебяибондасравниваетскрысобояминадопросесильварассказываетчтобудучивплен
упыталсяпокончитьссобойпрокусивкапсулусцианидомноядлишьспровоцировалостеоми
елитверхнейчелюстиилевойскуловойкостиврезультатечегоонвынужденноситьвставну
ючелюстьсметаллическойскулойнанейвскорепреступникуудаётсябежатьпрямоизштабк
вартирымибиуйтичерезподземныйходилинииметропреследуяегобондпонимаетсильвап
остоянноопережаетегоблагодарятомучтоконтролируеткомпьютернуюсетьбританскойра
зведкивсёвплотьдопленениязлементыпланасильвапытаетсяосуществитьпокушениенаж
изньмвовремязаседанияпарламентскогокомитетабондвперестрелкеспасаетмиувозитеё
вродовоепоместьебондоввшотландииособнякскайфоллсильвенамеренноставляютсле
двкомпьютернойсетими6

Профессионалы есть не только в MI6!

Упражнение 4.12.2

Разработайте программу, которая осуществляет замену в тексте одного символа на другой. Т.е. символы, которые не указаны в подстановке останутся без изменений. Таким образом у вас будет одно большое поле ввода (EditText) для ввода исходного текста, а также два маленьких поля ввода (также EditText) для ввода заменяемой буквы и буквы на которую заменяем. Кроме того, необходима кнопка для запуска процедуры замены (см рис.). Здесь желательно вспомнить основные принципы разработки программ под Андроид. Помним, что приложение падает при неправильном вводе (например, пустой символ для замены). Догадайтесь как исправить эту ошибку.



Приведем код приложения:

1.Макет - /res/layout/activity_main.xml

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="${relativePackage}.${activityClass}" >
    <EditText
        android:id="@+id/editText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:lines="10"
```

```
        android:hint="Введите текст"
        android:background="#cccccc"
        android:ems="10" >
        <requestFocus />
    </EditText>
    <EditText
        android:id="@+id/from_char"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentLeft="true"
        android:layout_below="@+id/editText"
        android:layout_marginLeft="28dp"
        android:layout_marginTop="22dp"
        android:background="#cccccc"
        android:ems="10"
        android:width="40dp" />
    <EditText
        android:id="@+id/to_char"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentLeft="true"
        android:layout_alignTop="@+id/from_char"
        android:layout_toRightOf="@+id/from_char"
        android:layout_marginLeft="100dp"
        android:width="40dp"
        android:background="#cccccc"
        android:ems="10" />
    <Button
        android:id="@+id/button1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignTop="@+id/from_char"
        android:layout_toRightOf="@+id/to_char"
        android:layout_marginLeft="50dp"
        android:onClick="replace"
        android:text="заменить" />
</RelativeLayout>
```

2. Класс активности: MainActivity.java

```
public class MainActivity extends Activity {
      TextView from_charTV,to_charTV;
      EditText edit_textET;
      String from_char,to_char,edit_text;
      @Override
      protected void onCreate(Bundle savedInstanceState) {
            super.onCreate(savedInstanceState);
            setContentView(R.layout.activity_main);
            from_charTV=(TextView)findViewById(R.id.from_char);
            to_charTV=(TextView)findViewById(R.id.to_char);
            edit_textET=(EditText)findViewById(R.id.editText);
      }
      public void replace(View v){
```

```
            from_char=from_charTV.getText().toString();
            to_char=to_charTV.getText().toString();
            edit_text=edit_textET.getText().toString();
            edit_text=edit_text.replaceAll(from_char, to_char);
            edit_textET.setText(edit_text);
      }
}
```

Упражнение 4.12.3

Расшифруйте сообщение, зашифрованное подстановочным шифром используя программу, разработанную в п.п. 4.12.1. Для более удобного криптоанализа зашифрованного сообщения добавьте в вашу программу еще одно текстовое поле (TextView), в которое выведите частотность встречающихся в первом текстовом поле букв. Подсказка - не забудьте из статистики убрать все символы не буквы (пробелы, символы, цифры и т.д.) иначе у вас исказятся частотности. Зашифрованный текст должен быть каким то широкоизвестным, например строфа из «Евгения Онегина» или стихотворение в прозе «Русский язык» И.С.Тургенева. Работа выполняется в комибинированном режиме – часть работы выполняется вручную (замена), часть – в автоматизированном (анализ частотности) режиме. Если в течение урока текст не удастся окончательно расшифровать, завершение работы можно задать в качестве домашнего задания.

Для решения этой задачи нужно добавить в предыдущее приложение дополнительную функцию расчета частотности, результат действия которой выводить в дополнтиельное TextView:

```
<TextView
        android:id="@+id/average"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_alignRight="@+id/from_char"
        android:layout_below="@+id/button1"
        android:layout_marginTop="15dp"
        android:lines="33"
        android:width="250dp"
        android:text="Таблица частотности(рус)" />
```

и дополнительный метод:

```
      public void calculate(View v){
            String s=edit_textET.getText().toString();
            s = s.toLowerCase().replaceAll("[^а-я]+", "");
            averageTV.setText("");
            for (int i = 0; i < s.length(); i++) {
                  char a = s.charAt(i);
                  if (s.substring(0, i).indexOf(a) != -1)
                        continue;
                  int ch = 0;
```

```
                    int k = i;
                    for (int j = k; j < s.length(); ch++) {
                          int ii = s.indexOf(a, j);
                          if (ii == -1)
                                break;
                          j = ii + 1;
                    }
                    averageTV.setText(averageTV.getText().toString() + "\n" + a + " "
+  String.format(  "%.4f", (double) ch / s.length()));
              }
       }
```

Задание 4.12.1

Доработать приложение из предыдущего упражнения, чтобы оно показывало все уже введенные замены, а так же частоту 5 наиболее часто встречающихся букв (*двубуквенных слов), для которых еще нет замены.

Благодарности

Компания Samsung Electronics выражает благодарность за участие в подготовке данного материала преподавателю IT ШКОЛЫ SAMSUNG Яценко Дмитрию Владимировичу.