Лабораторная работа №. 3. Многоалфавитные шифры

Многоалфавитные шифры (polyalphabetic ciphers). Слабость одноалфавитных шифров определяется тем, что их частотное распределение отражает распределение используемого алфавита. Шифр криптографически более надежен, если он имеет регулярное распространение, которое не предоставляет информацию криптоаналитику.

Один из способов сгладить распределение — объединить высокие распределения с низкими. Если буква *T* иногда зашифровывается как *a*, а иногда как *b*, и если буква *X* также иногда зашифровывается как *a*, а иногда как *b*, то высокая частота *T* сочетается с низкой частотой *X*, что приводит к более умеренному распределению для *a* и *b*. Два распределения могут быть объединены с использованием двух отдельных шифровальных алфавитов, например первый для четных символов в открытом тексте, второй для нечетных символов, что приводит к необходимости поочередного использования двух таблиц перевода, например перестановок, или по формулам:

$$p_1(a) = (3 \cdot a) \bmod 26 \text{ и } p_2(a) = (7 \cdot a + 13) \bmod 26.$$

3.1. Cifrul Vigenère

Метод шифрования, известный как «шифр Виженера», был ошибочно приписан Блезу де Виженеру в 19 веке, а на самом деле впервые был описан Джован Баттиста Белласо в его книге 1553 года *La cifra del. Sig*. Виженер создал очень похожий, но все же другой и более сильный шифр в 1586 году.

С другой стороны, шифр Виженера использует те же операции, что и шифр Цезаря. Число Виженера и ему подобные перемещают буквы, но, в отличие от Цезаря, его нельзя легко взломать за 26 комбинаций. Шифр Виженера использует множественный сдвиг. Ключ представляет собой не одно смещение, а несколько, порождаемых несколькими целыми числами k_i, где $0 \le k_i \le 25$, если взять за основу латинский алфавит из 26 букв. Шифрование выполняется следующим образом:

$$c_i = (m_i + k_i) \bmod 26.$$

Ключом шифрования может быть, например, $k = (5, 20, 17, 10, 20, 13)$ что заставит первую букву перемещаться на 5 позиций, $c_1 = m_1 + 5 \pmod{26}$, вторую на 20, $c_1 = m_1 + 20 \pmod{26}$ и т. д. до конца ключа, а затем циклически снова начиная с 5. Ключ обычно представляет собой слово, чтобы его было легче запомнить - ключ $k = (5, 20, 17, 10, 20, 13)$ соответствует слову «*furtun*». Метод множественного сдвига обеспечивает дополнительную защиту по двум причинам:

- первая причина - другие не знают длину ключа;
- вторая причина - количество возможных решений увеличивается с увеличением размера ключа; например, для длины ключа, равной 5, количество комбинаций, которые потребуются для полного перебора, составит $26^5 = 11\,881\,376$.

Дешифрование шифра Виженера аналогична шифрованию. Разница в том, что ключ вычитается из зашифрованного текста:

$$m_i = (c_i - k_i) \bmod 26.$$

Для упрощения процесса шифрования можно использовать таблицу 3.1, называемую *Tabula Recta*, которую использовал Vigenere. Здесь все 26 шифров расположены горизонтально и каждому шифру соответствует определенной букве в ключе, представленной в столбце слева от таблицы. Алфавит, соответствующий буквам открытого текста, находится в первой строке таблицы. Процесс шифрования прост – необходимо, имев букву $\boldsymbol{m_i}$ из сообщения и букву $\boldsymbol{k_i}$ из ключа, найти букву шифротекста $\boldsymbol{c_i}$, которая находится на пересечении строки $\boldsymbol{m_i}$ и столбца $\boldsymbol{k_i}$. Пример в таблице 3.1 показывает случай, когда $\boldsymbol{m_i}$=***M*** а $\boldsymbol{k_i}$ = ***H***, и в результате мы получаем $\boldsymbol{c_i}$ = ***T***.

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

Tabelul 3.1. *Tabula Recta pentru cifrul Vigenere*

Также можно действовать согласно уравнениям, определяющим математическую модель шифра:

$$c_i = m_i + k_i \pmod{26} \text{ și } m_i = c_i - k_i \pmod{26},$$

как показано в нижеследующем примере.

Пример.
Зашифровать с помощью шифра Виженера ключом ***K*** = ***SUPER*** сообщение «*Per aspera ad astra*».
Решение. Чтобы зашифровать или расшифровать сначала мы делаем следующее соответствие (кодируем алфавит):

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

Затем составляем и заполняем таблицу:

M	P	E	R	A	S	P	E	R	A	A	D	A	S	T	R	A
K	**S**	**U**	**P**	**E**	**R**	S	U	P	E	R	S	U	P	E	R	S
M	15	4	17	0	18	15	4	17	0	0	3	0	18	19	17	0
K	18	20	15	4	17	18	20	15	4	17	18	20	15	4	17	18
M+***K*** (mod 26)	7	24	6	4	9	7	24	6	4	17	21	20	7	23	8	18
C	H	Y	G	E	J	H	Y	G	E	R	V	U	H	X	I	S

Таким образом получаем криптограмму ***C***= *HYGEJHYGERVUHXIS*.

Для дешифрования делаем то же самое, применяя обратную формулу: $m_i = c_i - k_i \pmod{26}$.

C	H	Y	G	E	J	H	Y	G	E	R	V	U	H	X	I	S
K	S	U	P	E	R	S	U	P	E	R	S	U	P	E	R	S
C	7	24	6	4	9	7	24	6	4	17	21	20	7	23	8	18
K	18	20	15	4	17	18	20	15	4	17	18	20	15	4	17	18
M–K (mod 26)	15	4	17	0	18	15	4	17	0	0	3	0	18	19	17	0
M	P	E	R	A	S	P	E	R	A	A	D	A	S	T	R	A

Таким образом получаем исходное сообщение ***M*** = PERASPERAADASTRA.

3.2. Алгоритм шифрования Плейфера (Playfair)

3.2.1. Краткая история

Хотя алгоритм назван в честь барона Лайона Плейфера, он был изобретен его другом Чарльзом Уитстоном и впервые описан в документе от 26 марта 1854 г. Первоначально он был отклонен британским министерством иностранных дел, поскольку считался очень сложным для понимания. Когда Уитстон предложил показать, что за 15 минут он научит пользоваться алгоритмом 3-х мальчиков из 4-х из соседней школы, секретарь министерства иностранных дел ответил: «Да, очень даже возможно, но вы не сможете научить их быть хорошими дипломатами».

После создания алгоритма барон Плейфер убедил британское правительство принять этот алгоритм для официального использования, поэтому он назван в его честь, а не его создателя Уитстона. Алгоритм использовался британской армией во время англо-бурской войны в Южной Африке, а модифицированные версии также использовались британцами в Первой мировой войне и австралийской армией во Второй мировой войне.

С точки зрения современной криптографии алгоритм Плейфера устарел, даже примитивен. Любой современный персональный компьютер может найти (взломать) ключ и расшифровать сообщение за считанные секунды или доли секунды с помощью подходящего программного обеспечения. Некоторые из самых опытных криптоаналитиков или даже некоторые специалисты по кроссвордам могут взломать зашифрованное сообщение за считанные минуты, используя только карандаш и лист бумаги.

Хотя это устаревший алгоритм во всех отношениях, алгоритм Плейфера является одним из первых алгоритмов, использующих современные принципы блочных шифров. Изучение этого алгоритма может дать лучшее интуитивное понимание современной криптографии без использования сложных знаний по математике или теории чисел.

3.2.2. Общий обзор алгоритма

Шифрование алгоритмом Плейфера включает следующие шаги:

a) подготовка текста к шифрованию;
b) построение матрицы шифрования;
c) построение зашифрованного сообщения.

3.2.3. Детальный обзор процесса шифрования с примером

a) подготовка текста к шифрованию

Этот первый шаг включает в себя написание всех букв заглавными буквами, разбивая их на пары, без пробелов и знаков препинания. Все буквы «***J***» в тексте будут заменены на «***I***» (в нижеследующем примере буквы «***J***» нет).

Пусть имеется сообщение для шифрования:

> ***m*** *= Congress shall make no law respecting an establishment of religion, or prohibiting the free exercise thereof; or abridging the freedom of speech, or of the press; or the right of the people peaceably to assemble, and to petition the government for a redress of grievances.*

Сначала оно примет вид:

> ***CONGRESS SHALL MAKE NO LAW RESPECTING AN ESTABLISHMENT OF RELIGION OR PROHIBITING THE FREE EXERCISE THEREOF OR ABRIDGING THE FREEDOM OF SPEECH OR OF THE PRESS OR THE RIGHT OF THE PEOPLE PEACEABLY TO ASSEMBLE AND TO PETITION THE GOVERNMENT FOR A REDRESS OF GRIEVANCES***

Потом, разделив его на двухбуквенные группы (диграммы), получим:

> ***CO NG RE SS SH AL LM AK EN OL AW RE SP EC TI NG AN ES TA BL IS M EN TO FR EL IG IO NO RP RO HI BI TI NG TH EF RE EE EX ER CI SE TH ER EO FO RA BR ID GI NG TH EF RE ED OM OF SP EE CH OR OF TH EP RE SS OR TH ER IG HT OF TH EP EO PL EP EA CE AB LY TO AS SE MB LE AN DT OP ET IT IO NT HG OV ER NM EN TF OR AR ED RE SS OF GR IE VA NC ES***

Следующим шагом в подготовке текста к шифрованию является вставка буквы «***Q***», «***X***» или «***Z***» (самые редкие буквы в английской лексике) между каждой парой двойных букв. Например, слово «*FR **EE** DO M*» в приведенном выше примере станет «*FR **EX** ED OM*». Из-за трехкратного повторения буквы ***S*** между первыми 2 словами примера («*CO NG RE **SS** **S**H AL*») они будут переписаны как «*CO NG RE **SX** **SZ** **S**H AL L*».

Это правило двойных буквы было введено по двум причинам:

1. потому что двойные буквы очень распространены в английском языке, и это может помочь криптоаналитику;
2. чтобы уменьшить количество интуитивно понятных с первого взгляда слов в шифруемом сообщении (например, *miss*", "*missing*" и т. д.).

Последним шагом в подготовке текста к шифрованию является добавление дополнительной буквы, выбранной человеком, шифрующим сообщение, если на предыдущем шаге было получено нечетное количество букв. Окончательный текст нашего примера, готовый к шифрованию, будет таким:

> ***CO NG RE SX SZ SH AL LM MA KE NO LA WR ES PE CT IN GA NE ST AB LI SH ME NT OF RE LI GI ON OR PR OH IB IT IN GT FR EX EZ EX ER CI SE TH ER EO FO RA BR RI DG IN GT HE FR EX ED OM OF SP EX EC HO RO FT HE PR ES SO RT HE RI GH TO FT HE PE OP LE PE AC EA BL YT OA SX SE MB LE AN DT OP ET IT IO NT HE GO VE RN ME NT FO RA RE DR ES SO FG RI EV AN CE SB***

Примечание: В случае группы букв *SS* в группе *ES SO* буквы *Q*, *X* или *Z* не использовались, поскольку две буквы *S* не входят в одну и ту же двухбуквенную группу.

b) построение матрицы шифрования

На этом шаге из ключа шифрования будут удаляться повторяющиеся буквы, начиная с их второго появления, например, если ключ *k = dublura*, то он превратится в *dublra*.

Для шифрования нашего сообщения мы будем использовать в качестве ключа *k = First Amendment*, который станет после обработки *FIRST AMEND*.

После этого строится матрица шифрования, которая для 26-буквенного английского алфавита может быть размером 5x5. В общем случае, если алфавит языка, на котором написано шифруемое сообщение, имеет разное количество букв, то и матрица может быть другой, например 6x5, 5x6, 6x4, 4x6 и т.д., в которой включаются все (или почти все все) буквы алфавита. Если места для размещения всех букв алфавита не хватает - можно поступить, как было показано выше, т.е. заменить букву *J* на *I*, или можно удалить совсем 1-2 наиболее редко встречающуюся в этом языке буквы, а при расшифровании, интуитивно они будет восстановлены. Если в матрице ячеек больше, чем букв, свободные ячейки можно заполнить любыми символами.

После этого матрица будет заполнена согласно алгоритму:

- начиная с левого верхнего угла слева направо и сверху вниз записывается ключ из предыдущего шага;
- матрица будет дополнена остальными буквами алфавита, кроме буквы *J* или той что мы удалили, взятыми в алфавитном порядке.

Для нашего примера мы получим следующую матрицу шифрования:

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

Примечание: Чем длиннее используемый ключ, тем сложнее будет взламывать зашифрованный текст. Наиболее часто используемый метод использования длинного ключа заключался в запоминании коротких предложений (3-5 слов), которые легко запомнить.

c) построение зашифрованного сообщения

Буквенные пары в исходном тексте будут зашифрованы по алгоритму:

1. если две буквы находятся в разных строках и столбцах, каждая буква будет заменена буквой в той же строке, но в столбце другой буквы в текущей паре. Например, ***NP*** будет зашифровано как ***EQ***;
2. если обе буквы находятся на одной строке матрицы, каждая будет заменена следующей буквой в текущей строке; последняя буква в строке будет заменена первой буквой в той же строке. Например, пара ***IT*** будет зашифрована как ***RF***;
3. аналогично, если буквы находятся в одном столбце, каждая из них будет заменена буквой, находящейся в том же столбце, но строкой ниже; последняя буква в столбце будет заменена первой буквой того же столбца. Например, пара ***CW*** будет зашифрована в ***OI*** (поскольку ***W*** является последней в столбце и не имеет другой буквы под ней, она будет зашифрована первой буквой в том же столбце).

Используя исходное сообщение и ключ, обработанные на предыдущих шагах, а также матрицу из шага *b)*, мы получим следующий зашифрованный текст:

OWEHEGRYTYNQBVOAEMGDMQVBXINRXGKISMBEDNTFBLOF
NQENDSLIEGOFCRQMPIXEQCFCRFSMKRISGRDXGRGEOMRNSK
GEMPILFEGFSREKSMKRGNISGRNAWCLIRQGRMGCQIPIFGNXE
NRIQSFGNSRHKIUIFGNXGPQPAXGMBNMLVZSLMRYRNACPAM
DKDPQDRRFMWDSGNCPXASEENDSILFEEGETNRIQRBSRAXMDG
MFH

3.2.4. Дешифрование сообщения

Чтобы расшифровать сообщение с помощью алгоритма Плейфера, мы инвертируем все шаги шифрования. Разбиваем зашифрованный текст на пары (подготовительные действия из этапа шифрования нам не нужны):

OW EH EG RY TY NQ BV OAEM GD MQ VB XI NR XG KI SM BE DN TF
BL OF NQ EN DS LI EG OF CR QM PI XE QC FC RF SM KR IS GR DX
GR GE OM RN SK GE MP IL FE GF SR EK SM KR GN IS GR NA WC LI
RQ GR MG CQ IP IF GN XE NR IQ SF GN SR HK IU IF GN XG PQ PA XG
MB NM LV ZS LM RY RN AC PA MD KD PQ DR RF MW DS GN CP XA
SE EN DS IL FE EG ET NR IQ RB SR AX MD GM FH

По аналогии, имея тот же ключ шифрования, мы получаем ту же матрицу что и при шифровании:

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

Для дешифрования мы будем преобразовывать пары букв по тем же правилам, что и для шифрования, со следующими пояснениями:

1. если две буквы находятся в разных строках и столбцах, дешифрование производится точно так же, как и шифрование;
2. если обе буквы находятся на одной строке матрицы, каждая будет заменена предыдущей в текущей строке; первая буква в строке будет заменена последней буквой в той же строке;
3. если буквы находятся в одном столбце, каждая из них будет заменена буквой, находящейся в том же столбце, но на строку выше; первая буква в столбце будет заменена последней буквой в том же столбце.

По выполнении этих шагов мы получим;

CO NG RE SX SZ SH AL LM MA KE NO LA WR ES PE CT IN GA NE ST
AB LI SH ME NT OF RE LI GI ON OR PR OH IB IT IN GT FR EX EZ EX
ER CI SE TH ER EO FO RA BR RI DG IN GT HE FR EX ED OM OF SP
EX EC HO RO FT HE PR ES SO RT HE RI GH TO FT HE PE OP LE PE

AC EA BL YT OA SX SE MB LE AN DT OP ET IT IO NT HE GO VE RN ME NT FO RA RE DR ES SO FG RI EV AN CE SB

Сообщение теперь можно прочитать, если мы удалим пробелы между парами букв и добавим новые пробелы, в зависимости от используемого языка и логики сообщения:

CONGRESS SHALL MAKE NO LAW RESPECTING AN ESTABLISHMENT OF RELIGION OR PROHIBITING THE FREE EXERCISE THEREOF OR ABRIDGING THE FREEDOM OF SPEECH OR OF THE PRESS OR THE RIGHT OF THE PEOPLE PEACEABLY TO ASSEMBLE AND TO PETITION THE GOVERNMENT FOR A REDRESS OF GRIEVANCES.

Задание:

Задание 3.1. Реализовать алгоритм Плейфера на одном из языков программирования для сообщений на русском языке (33 буквы). Значения символов сообщения находятся между «A» и «Z», «a» и «z», и никакие другие значения не допускаются. Если пользователь введет другие значения – ему будет предложен правильный диапазон символов. Длина ключа должна быть не менее 7. Пользователь сможет выбрать операцию - шифрование или дешифрование, сможет ввести ключ, сообщение или криптограмму и получит криптограмму или расшифрованное сообщение. Завершающий этап добавления новых пробелов, в зависимости от используемого языка и логики сообщения - будет производиться вручную.

Задание 3.2. Реализовать алгоритм Виженера на одном из языков программирования для сообщений на русском языке (33 буквы). Буквы кодируются числами 0, 1, ... 32. Значения символов сообщения находятся между «A» и 'Z', 'a' и 'z' и никакие другие значения не допускаются. Если пользователь введет другие значения - будет предложен правильный диапазон символов. Длина ключа не должна быть меньше 7. Шифрование и дешифрование будет производиться по формулам математической модели, представленной выше. В сообщении сначала нужно убрать пробелы, потом все буквы будут заменены на заглавные. Пользователь сможет выбрать операцию - шифрование или дешифрование, сможет ввести ключ, сообщение или криптограмму и получит криптограмму или расшифрованное сообщение.

Примечание: Студенты, зарегистрированные списке группы под нечетным номером, выбирают задание 3.1, а под четным номером - задание 3.2. Срок выполнения - 2 недели.