Лабораторна робота №1

Класичні алгоритми симетричного та асиметричного шифрування. Шифр Цезаря. Квадрати Полібія. Шифри заміни.

Теоретичні відомості та приклади застосування.

Шифрування методом простої заміни

У загальному випадку шифр заміни здійснює перетворення, при якому літери або якісь інші фрагменти відкритого тексту замінюються відповідними фрагментами шифрованого тексту.

Найпростіший випадок шифрування заміною полягає в тому, що знаки відкритого тексту, записані в одному (первинному) алфавіті, заміняють знаками, що взято із іншого (вторинного) алфавіту, у відповідності з наперед установленим правилом.

Якщо використовується один і той же вторинний алфавіт, то шифр заміни називають моноалфавітним. Якщо вторинних алфавітів декілька, то шифр називають багатоалфавітним.

Нехай шифруються повідомлення за допомогою первинного алфавіту. Тоді кожній букві вихідного алфавіту зіставляється деяка множина символів (шифрозаміни), наприклад для українського алфавіту А ↔ M_1=$M_А$, Б ↔ M_2=$M_Б$, ..., Я ↔ M_{33}=$M_Я$ (рис. 1.1)

А	Б	...	Я
$M_А$	$M_Б$	...	$M_Я$

Рис. 1.1. Таблиця замін

Шифрозаміни обираються таким чином, щоб будь-які дві множини не містили однакових елементів $M_i \cap M_j = \{\emptyset\}$.

Наведена таблиця є ключем шифру заміни, за допомогою якої можна здійснити як шифрування, так і розшифрування.

Наприклад, таблиця простої заміни (кожна множина M_i складається тільки з одного елемента) може мати наступний вигляд:

Символи первинного алфавіту	а	б	в	г	д	е	ж	з	і	ї	л	м	н	о	п	р	с	...
Символи вторинного алфавіту (заміна)	s	p	x	l	r	z	i	m	a	y	e	d	w	t	b	g	v	...

Рис. 1.2. Таблиця простої заміни.

Шифрограма для тексту «перемога», відповідно до таблиці простої заміни (рис.1.2) буде наступною – «bzgzdtls».

Квадрати Полібія (Полібіанський квадрати)

Одним з перших методів моноалфавітного шифрування був запропонований грецьким істориком Полібієм (II ст. до н. е). Він використовував квадратну таблицю розміром 5×5, заповнену випадково

буквами грецького алфавіту (*сучасна грецька абетка складається із 24 літер*) – *Полібіанський квадрат* (рис. 1.3).

λ (лямбда)	ν (ні)	θ (тета)	σ (сігма)	χ (хі)
ρ (ро)	π (пі)	υ (іпсилон)	α (альфа)	ι (йота)
μ (мі)	η (ета)	δ (дельта)	ϕ (фі)	γ (гама)
ψ (псі)	ξ (ксі)	β (бета)	ω (омега)	ο (омікрон)
κ (каппа)	ε(епсилон)		ζ (дзета)	τ (тау)

λ	ν	θ	σ	χ
ρ	π	υ	α	ι
μ	η	δ	ϕ	γ
ψ	ξ	β	ω	ο
κ	ε		ζ	τ

Рис. 1.3. Полібіанський квадрат, заповнений буквами грецького алфавіту + пробіл

Опис алгоритму. При шифруванні початкового повідомлення в цьому Полібіанському квадраті знаходили чергову букву тексту і записували в зашифрований текст букву, розташовану нижче за неї в тому ж самому стовпці. Якщо буква тексту опинялася в нижньому рядку таблиці, то для зашифрованого тексту брали верхню букву з того ж самого стовпця.

Наприклад у відповідності до таблиці–ключа на рис. 1.3, для слова **εορτή** (свято) отримаємо зашифрований текст **ντμχξ**.

Зауважимо, що для інших алфавітів використовуються квадрати інших розмірів. Наприклад для українського алфавіту (без літери ґ) використовується квадрат $\mathbf{6 \times 6}$ (рис. 1.4). При цьому до 32 літер алфавіту додають чотири поширених основних символи: _ (пробіл); . (крапка); , (кома); ' (апостроф).

а	б	в	г	д	е
є	ж	з	и	і	ї
й	к	л	м	н	о
п	р	с	т	у	ф
х	ц	ч	ш	щ	ю
я	ь	_	.	,	'

$\Rightarrow$

ь	й	ю	і	ш	з
ц	є	ф	д	т	в
р	а	о	,	м	_
к	я	ї	щ	и	ч
ж	х	е	у	г	с
б	п	'	н	.	л

Рис. 1.4. Полібіанський квадрат для українського алфавіту.

Приклад 1.1. За допомогою Полібіанського квадрату, що наведений на рис. 1.4 зашифрувати текст із 33 символів: "Все йде, все минає, і краю немає.".

■Для зручності складаємо таблицю замін.

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	26	27	28	29	30	31	32
В	с	е	_	й	д	е	,	_	в	с	е	_	м	и	н	а	є	,	_	і	_	к	р	а	ю	_	н	е	м	а	є	.
_	л	'	ч	є	,	'	щ	ч	_	л	'	ч	и	г	і	я	а	щ	ч	д	ч	ж	к	я	ф	ч	і	'	и	я	а	ш

Тоді результати шифрування матимуть наступний вигляд:

_л'чє,'щч_л'чигіящчдчжкяфчі'ияаш ■

Змінивши випадково заповнення полібіанського квадрата на інше розміщення символів, отримаємо на виході зовсім інший зашифрований текст, тобто розміщення символів у квадраті є ключем до шифрування/дешифрування різних повідомлень.

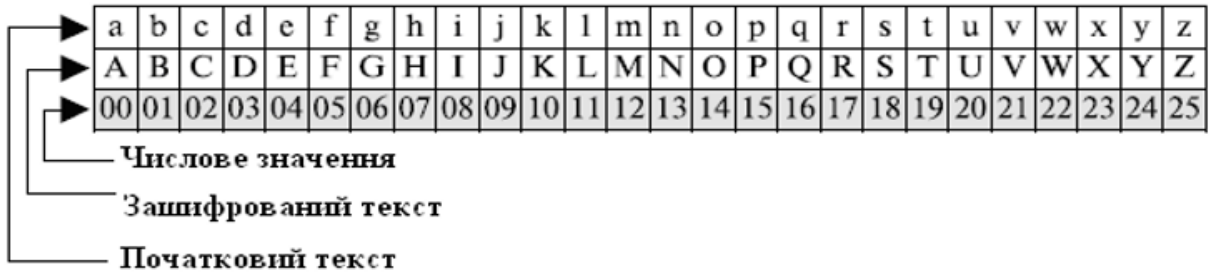
Моноалфавітна звичайна заміна Цезаря (шифр Цезаря)

Римський імператор Гай Юлій Цезар (I ст. до н. е) використовував у своєму військовому та особистому листуванні шифр, суть якого полягала у заміні кожної літери повідомлення на одну з інших літер того ж самого алфавіту 26-значного латинського алфавіту.

Принцип дії полягає в тому, щоб циклічно зсунити алфавіт, а ключ — це кількість літер, на які робиться зсув.

Зауважимо, що шифр Цезаря має замало ключів — на одиницю менше, ніж літер в абетці. Тому перебрати усі ключі не складає особливої роботи. Дешифрування з одним з ключів дасть нам вірний відкритий текст.

Припустимо, що початковий текст складається з маленьких букв (від a до z) і зашифрований текст складається із заголовних букв (від A до Z). Зіставимо кожній букві числове значення його порядковий номер (рис .1.5).



00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
А	Б	В	Г	Ґ	Д	Е	Є	Ж	З	И	І	Ї	Й	К	Л	М	Н	О	П	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ь	Ю	Я
Г	Ґ	Д	Е	Є	Ж	З	И	І	Ї	Й	К	Л	М	Н	О	П	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ь	Ю	Я	А	Б	В

Рис. 1.5. Таблиці замін для шифру Цезаря для англійського (ключ k=0) та українського (ключ k=3) алфавітів .

Зауваження. Якщо зіставити кожному символу алфавіту його порядковий номер (нумеруючи з 0), то шифрування й дешифрування можна виразити формулами:

$$Y = (X + k) \bmod n \quad \text{і} \quad X = (У - k) \bmod n, \text{ де:}$$

X — символ відкритого тексту,
Y — символ шифрованого тексту,
n — потужність алфавіту, k — ключ.

Приклад 1.2 За допомогою шифру Цезаря з ключем k=3, зашифрувати: словосполучення «**ШИФР ЦЕЗАРЯ**» з використанням таблиці замін для українського алфавіту; словосполучення «***veni vidi vici***» з використанням таблиці замін для англійського алфавіту.

■Для цього зрушимо алфавіт так, щоб він починався з четвертої букви (Г).

00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
А	Б	В	Г	Ґ	Д	Е	Є	Ж	З	И	І	Ї	Й	К	Л	М	Н	О	П	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ь	Ю	Я
Г	Ґ	Д	Е	Є	Ж	З	И	І	Ї	Й	К	Л	М	Н	О	П	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ь	Ю	Я	А	Б	В	

Отже, отримуємо відповідність Г↔А, Ґ↔Б, Д↔В, і т. д.

Використовуючи цю схему, відкритий текст «ШИФР ЦЕЗАРЯ» перетворюється на «ЮЙЧУ ЩЗЇГУВ». Для того, щоб одержувач повідомлення міг відновити вихідний текст, необхідно повідомити йому, що ключ — 3.

$a \to d$	$d \to g$	$g \to j$	$j \to m$	$m \to p$	$p \to s$	$s \to v$	$v \to y$	$y \to b$
$b \to e$	$e \to h$	$h \to k$	$k \to n$	$n \to q$	$q \to t$	$t \to w$	$w \to z$	$z \to c$
$c \to f$	$f \to i$	$i \to l$	$l \to o$	$o \to r$	$r \to u$	$u \to x$	$x \to a$	

Відкритий текст «veni vidi vici» (укр. переклад «Прийшов, побачив, переміг»), в зашифрованому вигляді буде записано як « yhql ylgl ylfl».■

Система шифрування Цезаря є моноалфавітним підстановким, яка шифрує n-граму (x0, x1, x2, ..., xn-1) початкового тексту в n-граму (y0, y1, y2, ..., yn-1) зашифрованого тексту згідно з таким правилом:

$$y_j = E_k(x_j),\ j = \overline{0, n-1}; \qquad E_k : i \to (i+k) \bmod m,\ 0 \le k < m, \qquad (1.3)$$

де i – числовий код букви початкового тексту; i + k – числовий код відповідної букви зашифрованого тексту.

Система шифрування Цезаря утворює за своєю сутністю сімейство моноалфавітних підставлянь для вибраних випадково значень ключа k, причому $0 \le k < m$.

Перевагою системи шифрування Цезаря є простота шифрування вхідного повідомлення та дешифрування зашифрованого тексту. До недоліків системи шифрування Цезаря необхідно віднести такі:

- підстановки, що виконується згідно з системою шифрування Цезаря, не маскують частот появи різних букв початкового тексту;
- зберігається алфавітний порядок в послідовності замінюваних букв; при зміні значення *k* змінюються тільки початкові позиції такої послідовності;
- кількість можливих ключів *k* є надто малою;
- шифр Цезаря легко розкривається на підставі аналізу частот появи букв у зашифрованому тексті.

Криптоаналітична атака проти системи моноалфавітної заміни починається з підрахунку частот появи символів: визначається кількість появ кожної букви в зашифрованому тексті. Потім отриманий розподіл частот букв у зашифрованому тексті порівнюється з розподілом частот букв у алфавіті початкових повідомлень, наприклад, в українському алфавіті. Буква з найбільшою частотою появи в зашифрованому тексті замінюється на букву з найбільшою частотою появи в українській мові і так далі. Ймовірність

успішного розкриття системи шифрування Цезаря підвищується із збільшенням довжини зашифрованого тексту.

Концепція, закладена в систему шифрування Цезаря, виявилася надзвичайно плідною, про що свідчать її численні модифікації.

Афінна система підстановок (заміни) як розвиток шифру Цезаря

У системі шифрування Цезаря використовувалися тільки адитивні властивості множини цілих чисел $\bar{Z}_m$. Застосовуючи одночасно операції додавання та множення за модулем m над цілими числами із $\bar{Z}_m$, отримують метод шифрування, що називають афінною системою підстановки Цезаря.

У цьому методі буква початкового тексту, яка відповідає числу t, замінюється на букву, що відповідає числовому значенню (a·t + b) за модулем m. Тут ключами виступають числа a і b. Необхідно відзначити, що перетворення шифрування – дешифрування є взаємно однозначним відображенням на множині цілих чисел тільки в тому випадку, якщо найбільший спільний дільник чисел a і m, що позначається як НСД(a, m), дорівнює одиниці, тобто a і m мають бути взаємно простими числами.

Наприклад, для англійського алфавіту m = 26 при a = 3, НСД(3, 26) = 1, b = 5, отримуємо таку відповідність між числовими кодами букв:

t	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
$p=a\cdot t+b$	5	8	11	14	17	20	23	26	29	32	35	38	41	44	47	50	53	56	59	62	65	68	71	74	77	80
$p \bmod m$	5	8	11	14	17	20	23	0	3	6	9	12	15	18	21	24	1	4	7	10	13	16	19	22	25	2

Перетворюючи числа в букви англійського алфавіту, отримуємо таку відповідність для букв початкового тексту і зашифрованого тексту:

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
a	b	c	d	e	f	q	h	i	j	k	l	м	n	o	p	q	r	s	t	u	v	w	x	y	z
5	8	11	14	17	20	23	0	3	6	9	12	15	18	21	24	1	4	7	10	13	16	19	22	25	2
f	i	l	o	r	u	x	a	d	q	j	м	p	s	v	y	b	e	h	k	n	q	т	w	z	c

Наприклад, початкове повідомлення encryption перетвориться в такий зашифрований текст: rslezykdvs.

Приклад 1.3. Зашифруйте текст "***Система шифрування Цезаря.***" за допомогою афінної системи підстановок Цезаря з параметрами $m = 36$, $a = 7$ і $b = 13$ НСД(7, 36) = 1 стосовно українського алфавіту.

■ Будуємо таблицю відповідностей між числовими кодами символів.

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	26	27	28	29
30	31	32	33	34	35

$\Rightarrow$

13	20	27	34	5	12
19	26	33	4	11	18
25	32	3	10	17	24
31	2	9	16	23	30
1	8	15	22	29	0
7	14	21	28	35	6

Отримуємо відповідність для початкового і зашифрованого тексту:

а	б	в	г	д	е
є	ж	з	и	і	ї
й	к	л	м	н	о
п	р	с	т	у	ф
х	ц	ч	ш	щ	ю
я	ь	_	.	,	'

⇒

к	с	ш	,	е	й
р	ч	.	д	ї	п
ц	_	г	і	о	х
ь	в	и	н	ф	я
б	з	м	у	ю	а
ж	л	т	щ	'	є

Результати шифрування тексту "*Система шифрування Цезаря.*":

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**
С	и	с	т	е	м	а		ш	и	ф	р	у	в	а	н	н	я		Ц	е	з	а	р	я	.
и	д	и	н	й	і	к	т	у	д	я	в	ф	ш	к	о	о	ж	т	з	й	.	к	в	ж	щ

Зашифрований текст: "*идинйіктудявфшкоожтзй.квжщ*".

Система шифрування Цезаря з ключовим словом

Особливість цієї системи полягає у використанні ключового слова для зсуву і зміни порядку символів у алфавіті підстановки.

Виберемо довільне k, $0 \le k \le 25$ і слово або коротку фразу, бажано без однакових букв, як *ключове слово.* Нехай вибрано ключове слово *decryption* і число $k = 5$. Ключове слово записуємо під буквами алфавіту, починаючи з букви, числовий код якої співпадає з вибраним числом k.

0	1	2	3	4	5					10					15					20					25
a	b	c	d	e	f	q	h	i	j	k	l	м	n	o	p	q	r	s	t	u	v	w	x	y	z
					d	e	c	r	y	p	t	i	o	n											

Букви англійського алфавіту, що залишилися, записуємо перед і після ключового слова за алфавітом.

					5																				
a	b	c	d	e	f	q	h	i	j	k	l	м	n	o	p	q	r	s	t	u	v	w	x	y	z
a	b	f	q	h	d	e	c	r	y	p	t	i	o	n	j	k	l	м	q	s	u	v	w	x	z

Отримали таблицю замін.

Приклад 1.4. Зашифруйте текст "*Все йде, все минає.*" з використанням системи шифрування Цезаря з ключовим словом *секундомір* стосовно українського алфавіту $m = 36$ та ключем $k = 7$. Отримаємо таблицю замін:

а	б	в	г	д	е
є	ж	з	и	і	ї
й	к	л	м	н	о
п	р	с	т	у	ф
х	ц	ч	ш	щ	ю
я	ь	_	.	,	'

⇒

ю	я	ь	_	.	,
'	*с*	*е*	*к*	*у*	*н*
д	*о*	*м*	*і*	*р*	а
б	в	г	є	ж	з
и	ї	й	л	п	т
ф	х	ц	ч	ш	щ

Результати його шифрування матимуть такий вигляд:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
В	с	є	_	й	д	е	,	_	в	с	е	_	м	и	н	а	є	.
ь	г	,	ц	д	.	,	ш	ц	ь	г	,	ц	і	к	р	ю	'	ч

Шифрувальні таблиці Трісемуса (Трітеміуса)

Шифр Трісемуса (лат. Iohannes Trithemius, XV-XVI ст.) вважають удосконаленим шифром Цезаря. За алгоритмом шифрування кожен символ повідомлення замінюють на символ, розміщений лівіше за цей на певну величину (крок зміщення). Крок зміщення є змінним й залежить від додаткових умов. Наприклад, можна задати закон зміщення (рівняння шифрування) у вигляді лінійної функції від позиції літери, яку шифрують. Сама функція має гарантувати отримання цілочисельних значень, та мати обернену, що також забезпечує цілочисельність.

Наприклад обчислення кроку зміщення k можна задати формулами

$$k = Ap + B, \qquad k = Ap^2 + Bp + C$$

де p – позиція літери в повідомленні; A, B, C – ключі.

Рівняння шифрування для шифру Трітеміуса має такий вигляд:

$$L = (m + k)\ mod\ N$$

де L – номер зашифрованої букви в алфавіті; m – номер позиції чергової букви відкритого тексту в алфавіті; k – крок зміщення (функціональна залежність від позиції літери в повідомленні); N – потужність алфавіту.

Наведемо інший варіант побудови таблиці замін для шифру Трітеміуса.

Для такого шифру заміни, використовувалася таблиця для запису букв алфавіту і ключового слова. У таблицю спочатку вписувалося рядками ключове слово, причому букви, що повторювалися, відкидалися. Потім таблиця доповнювалася буквами алфавіту, що не увійшли до неї, за порядком. Наприклад для ключового слова *мікроелектроніка,* після відкидання однакових букв в таблицю замін буде вписано *мікроелтна.*

Спосіб шифрування запозичений із методу полібіанського квадрата. При шифруванні чергову літеру відкритого тексту знаходили у шифрувальній таблиці, а у криптограму записували літеру, розташовану рядком нижче в тому ж стовпчику. Якщо літера знаходилась у нижньому рядку таблиці, то для криптограми брали саму верхню літеру з того ж стовпчика.

Приклад 1.5. Зашифрувати текст «таблиця Трісемуса» за допомогою шифрувальних таблиць Трісемуса та ключового слова *мікроелтна.*

а	б	в	г	д	е
є	ж	з	и	і	ї
й	к	л	м	н	о
п	р	с	т	у	ф
х	ц	ч	ш	щ	ю
я	ь	_	.	,	'

⇒

м	і	к	р	о	е
л	т	н	а	б	в
г	д	є	ж	з	и
ї	й	п	с	у	ф
х	ц	ч	ш	щ	ю
я	ь	_	.	,	'

Результати шифрування: «*шлтпжьмкшцзчвсщчл*».

Такі табличні шифри називаються монограмними, оскільки шифрування виконується по одному символу. Трісемус першим відзначив, що шифрувальні таблиці дають змогу шифрувати відразу по два символи. Такі шифри називаються біграмними.

Біграмний шифр Плейфера (Playfair) або квадрат Плейфера.

Шифр Плейфера (винайдено в 1854 р.) є найбільш відомим біграмним шифром заміни. Він застосовувався Великобританією під час Першої світової війни. Основою шифру Плейфера є шифрувальна таблиця з випадково розташованими символами алфавіту початкових повідомлень.

Загалом структура шифрувальної таблиці системи Плейфера повністю аналогічна структурі шифрувальної таблиці Трісемуса.

Опишемо процедуру шифрування на прикладі таблиць прикладу 1.5.

Приклад 1.6. Зашифрувати текст «таблиця Трісемуса.» за допомогою системи Плейфера та ключового слова *мікроелтна*

1. Вхідний текст початкового повідомлення розбиваємо на пари символів (біграми). «та–бл–иц–я –Тр–іс–ем–ус–а.»Текст повинен мати парну кількість символів і в ньому не повинно бути біграм, що містять два однакові символи. Якщо ці вимоги не виконано, то початковий текст потрібно модифікувати.

2. Послідовність біграм початкового тексту перетвориться за допомогою шифрувальної таблиці в послідовність біграм зашифрованого тексту за таким правилом:

1) Якщо обидва символи біграми початкового тексту не потрапляють на один рядок або стовпець, наприклад ***–иц–***, тоді знаходять символи в кутах прямокутника, що визначається даною парою символів. У нашому прикладі це букви ***–юд–***. Пара букв ***–иц–*** відображається в пару ***–юд–.*** Послідовність символів біграми зашифрованого тексту повинна бути дзеркально розташованою за відношенням до послідовності символів біграми початкового тексту.

2) Якщо обидва символи біграми початкового тексту належать одному стовпцю таблиці, то символами зашифрованого тексту вважаються символи, які знаходяться під ними. Наприклад, біграма **нп** дає біграму зашифрованого тексту **єч**, біграма **са** дає біграму зашифрованого тексту **шж**. Якщо при цьому символ початкового тексту знаходиться в нижньому рядку, то для зашифрованого тексту береться відповідний символ з верхнього рядка того ж самого стовпця. Наприклад, біграма **дь** дає біграму зашифрованого тексту **йі**.

3) Якщо обидва символи біграми початкового тексту належать одному рядку таблиці, то символами зашифрованого тексту вважаються символи, які знаходяться праворуч від них. Наприклад, біграма **сй** дає біграму зашифрованого тексту **уп**. Якщо при цьому символи початкового тексту знаходиться в крайньому правому стовпці, то для шифру беруть відповідний символ з лівого стовпця в тому ж самому рядку. Наприклад, біграма **сф** дає біграму тексту **уї**.

Таким чином зашифрований текст «нб–вт–юд–ь.–іа–йр–мі–фу–жр».

Необхідно зазначити, що шифрування біграмами різко підвищує стійкість шифрів до розкриття.

Біграмний двотабличний шифр або шифр Вітстона (Уитстона)

Цей шифр винайдений у 1854 році англійцем Чарльзом Витстоном. Такий метод використовує дві прямокутні таблиці однакового розміру (по можливості, якнайближчі до квадрату), в кожній з яких випадковим способом розміщено один і той же алфавіт.

Процедура шифрування.

1. Відкритий текст розбивають на пари знаків – біграми. Перший знак біграми відкритого тексту фіксується у першій таблиці, другий знак біграми – у другій. Між зафіксованими знаками вибудовується уявний прямокутник. Одна діагональ цього прямокутника з'єднує знаки біграми відкритого тексту, друга діагональ дає результуючу біграму до криптограми. Перший знак результуючої біграми теж прочитується із першої таблиці, другий знак біграми – із другої таблиці.
2. Якщо знаки відкритого тексту потрапили в один і той же рядок, то і біграма криптограми береться з того ж рядка. Перший знак біграми криптограми береться із першої таблиці у стовпчику, номер якого такий же, як і номер стовпчика другого знаку біграми відкритого тексту. Другий знак біграми криптограми береться із другої таблиці у стовпчику, номер якого такий же, як і номер стовпчика першого знаку біграми відкритого тексту.

Приклад 1.7 При використанні алфавіту, який складається із десяти цифр, крапки та пропуску, шифрувальна таблиця може бути такою:

Таблиця 1			Таблиця 2		
2	7	.	0	2	7
6	0	3	_	.	4
1	4	9	6	8	5
_	5	8	3	1	9

Рис. 1.6. Приклад шифрувальних таблиць Вітстона.

У відповідності з цією таблицею біграму «78» буде зашифровано як «42», біграму «42» – як «78», біграму «59» – як «81» і т.д. А для повідомлення «2.718_3.14» одержимо криптограму «6252330465».

Перевагою біграмного двотабличного шифру у порівнянні з біграмним шифром Плейфейра є можливість використання біграм з однаковими знаками.

Азбука Морзе

Азбука Морзе (амер. Семюель Морзе представив у 1838 році) або морзянка – це особливий спосіб кодування знаків, який шифрує букви алфавіту, цифри і розділові знаки за допомогою послідовності сигналів: довгих («тире») і коротких («крапок»).

Всі знаки в коді Морзе утворюють так звану азбуку Морзе (рис.1.7).

Існує щонайменше два варіанти азбуки Морзе українською мовою, які незначно відрізняються. Один з них використовується Пластом, інший описаний у Додатку 27 до Регламенту аматорського радіозв'язку України. Приклад цих таблиць можна знайти за посиланням https://uk.wikipedia.org/wiki/Азбука_Морзе .

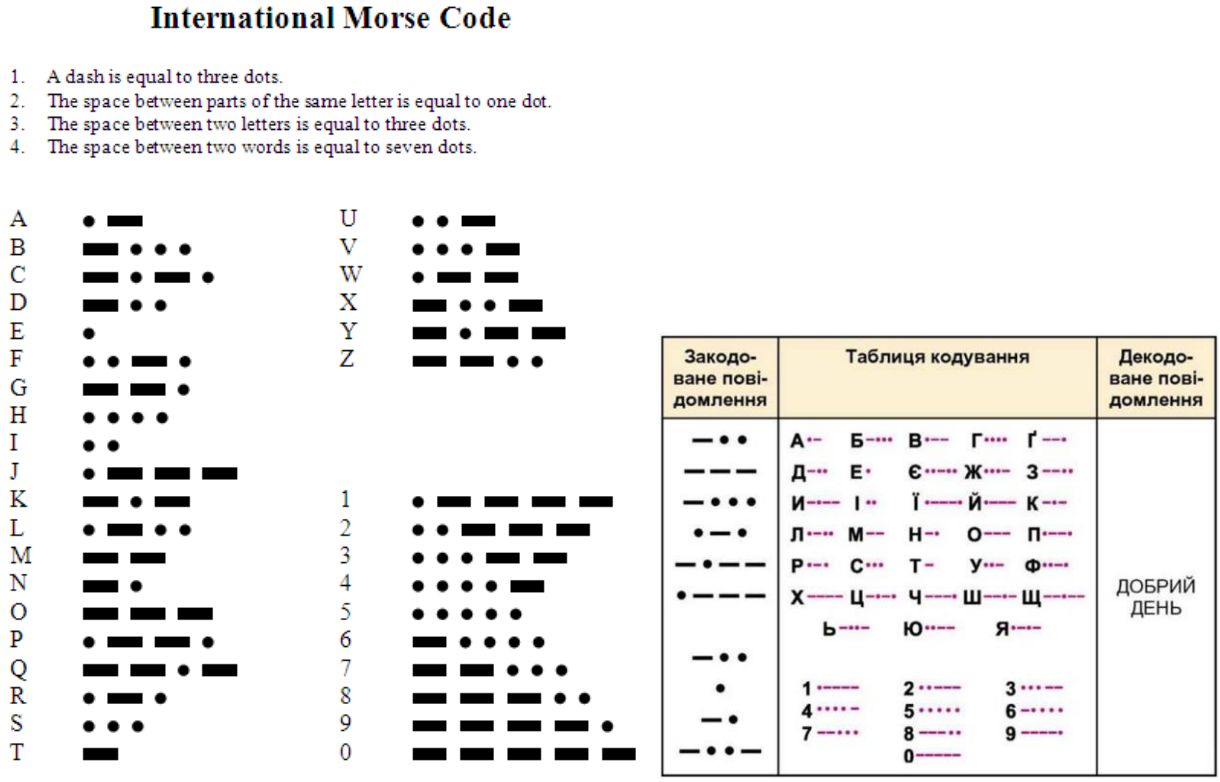


Рис. 1.7. Приклад шифрувальних (кодових) таблиць азбуки Морзе.

Система омофонів

Система омофонів (homophones) забезпечує простий захист інформації від криптоаналітичних атак, які базуються на підрахунку частот появи букв у зашифрованому тексті.

Дані про розподіли ймовірностей появи букв в українському і англійському текстах наведено в табл. 1, 2. Букви в таблицях вказані в порядку убування ймовірності їх появи в тексті. Наприклад, українська буква **е** трапляється в 13 разів частіше, ніж буква **є**, а англійська буква **e** трапляється в 123 рази частіше, ніж буква **z**.

Шифруючи букву початкового повідомлення, вибирають випадково одну з її замін. Заміни (часто їх називають омофонами) можуть бути представлені трицифровими числами від 000 до 999. Наприклад, в українському алфавіті букві **а** присвоюються 86 випадкових номерів, буквам **е** і **р** – по 47 номерів, а букві **ф** – всього 2 номери. Якщо омофони (заміни) присвоюються випадково різним появам однієї і тієї ж букви, тоді кожен омофон з'являється в зашифрованому тексті рівноймовірно.

Табл. 1. Розподіл ймовірностей появи букв у українських текстах

Буква	Ймовірність	Буква	Ймовірність	Буква	Ймовірність	Буква	Ймовірність
о	0,104	с	0,039	з	0,020	х	0,010
а	0,086	л	0,039	й	0,020	щ	0,009
і	0,068	к	0,037	б	0,018	ж	0,008
н	0,061	у	0,037	я	0,017	ц	0,007
в	0,054	д	0,030	г	0,015	ю	0,007
т	0,051	п	0,030	'	0,015	ї	0,007
е	0,047	м	0,027	ч	0,014	є	0,004
р	0,047	ь	0,025	ш	0,011	ф	0,002

Табл. 2. Розподіл ймовірностей появи букв у англійських текстах

Буква	Ймовірність	Буква	Ймовірність	Буква	Ймовірність
e	0,123	l	0,040	b	0,016
т	0,096	d	0,036	g	0,016
a	0,081	c	0,032	v	0,009
o	0,079	u	0,031	k	0,005
n	0,072	p	0,023	q	0,002
i	0,071	f	0,023	x	0,002
s	0,066	м	0,022	j	0,001
r	0,060	w	0,020	z	0,001
h	0,051	y	0,019		

Система омофонів є моноалфавітною, хоча при цьому букви початкового повідомлення мають декілька замін. Кількість замін береться пропорційною ймовірності появи букви у вхідному тексті.

№ п/п	А	Б	В	...	М	...	О	...	Р	...	Я
1	311	128	175	...	037	...	248	...	064	...	266
2	357	950	194	...	149	...	267	...	189	...	333
...	...	...	...	...	...	...	...	...	...	...	...
16	495	990	199	...	349	...	303	...	374	...	749
...	...		...	...	...	...	...	...	...	...	...
20	519		427	...	760	...	306	...	469	...	845
...	...		...	...	...	...	...	...	...		
32	637		524	...	777	...	432	...	554		
...	...		...				...	...	...		
45	678		644				824	...	721		
...	...						...	...	...		
47	776						828	...	954		
...	...						...				
80	901						886				
...							...				
110							903				

Рис. 1.8. Приклад шифрувальної таблиці для системи омофонів.

Наприклад за таблицею на рис. 1.8 текст «БРАМАМАМБА» перетвориться на шифрограму «128 064 311 037 357 149 495 349 950 519».

При такому підході до формування зашифрованого тексту простий підрахунок частот вже нічого не дає криптоаналітику.

Координатні заміни

В координатних замінах знаки алфавіту використовуються для позначень координат шифрувальної таблиці. Якщо алфавіт має N знаків і мова йде про двохкоординатну заміну, то шифрувальна таблиця має форму квадрата розміром $N*N$. В окремих комірках таблиці випадковим способом розміщують всі N можливих пар знаків, а вертикалі та горизонаталі таблиці позначають знаками, розташованими в алфавітному порядку. Відкритий текст розбивають на пари знаків – біграми. Перший знак біграми відкритого тексту використовується як індекс рядка, другий знак біграми – як індекс стовпчика. На їх перетині знаходиться результуюча біграма до криптограми. *Наприклад*, при використанні алфавіту «0, 1, 2» шифрувальна таблиця може мати такий вигляд:

	0	1	2
0	10	21	01
1	00	20	11
2	12	02	22

Для такої шифрувальної таблиці повідомлення «1020110221» перетворюється на криптограму «0012200102».

Суть ускладненого координатного методу, полягає в тому, що таблиці з числами ставиться у відповідність таблиця з алфавітом, тоді кожній літері у відповідність ставляють числа, приклад:

	0	1	2		0	1	2
0	10	21	01	0	А	Б	В
1	00	20	11	1	Г	Д	Е
2	12	02	22	2	Ж	Й	К

Тоді, наприклад, слово БАЙДА перетвориться на «21 10 02 20 10».

Шифр Віженера

Шифр Віженера — поліалфавітний шифр, який у якості ключа використовує слово або фразу. Був створений Блезом де Віженером, французьким математиком шістнадцятого сторіччя.

Шифр Віженера складається з послідовності декількох шифрів Цезаря з різними значеннями зсуву, що визначаються літерами ключового слова. Кожна літера відкритого тексту зсувається вперед на позицію відповідної літери ключа. Якщо ключове слово менше за повідомлення, то воно циклічно повторюється.

Приклад 1.8 Повідомлення ATTACK AT DAWN зашифруйте ключем LEMON.

	A	T	T	A	C	K	A	T	D	A	W	N
	L	E	M	O	N	L	E	M	O	N	L	E
	0	19	19	0	2	10	0	19	3	0	22	13
+	11	4	12	14	13	11	4	12	14	13	11	4
	11	23	5	14	15	21	4	5	17	13	7	17
	L	X	F	O	P	V	E	F	R	N	H	R

В результаті чого отримаємо шифротекст LXFOPVEFRNHR.

Для зашифровування може використовуватися й таблиця, яка отримала назву таблиця Віженера рис.1.9. У загальному випадку таблиця Віженера складається з алфавіту, циклічно зміщеного на один символ ліворуч.

Під час зашифрування кожна літера повідомлення замінюється на літеру, що знаходиться на перетині літер першого рядка (алфавіт повідомлення) і першого стовпчика (алфавіт ключа) в таблиці Віженера.

Приклад 1.9 Повідомлення PURPLE, зашифроване ключем SMART за допомогою таблиці Віженера, перетвориться у шифротекст HGRGEW(рис.1.9).

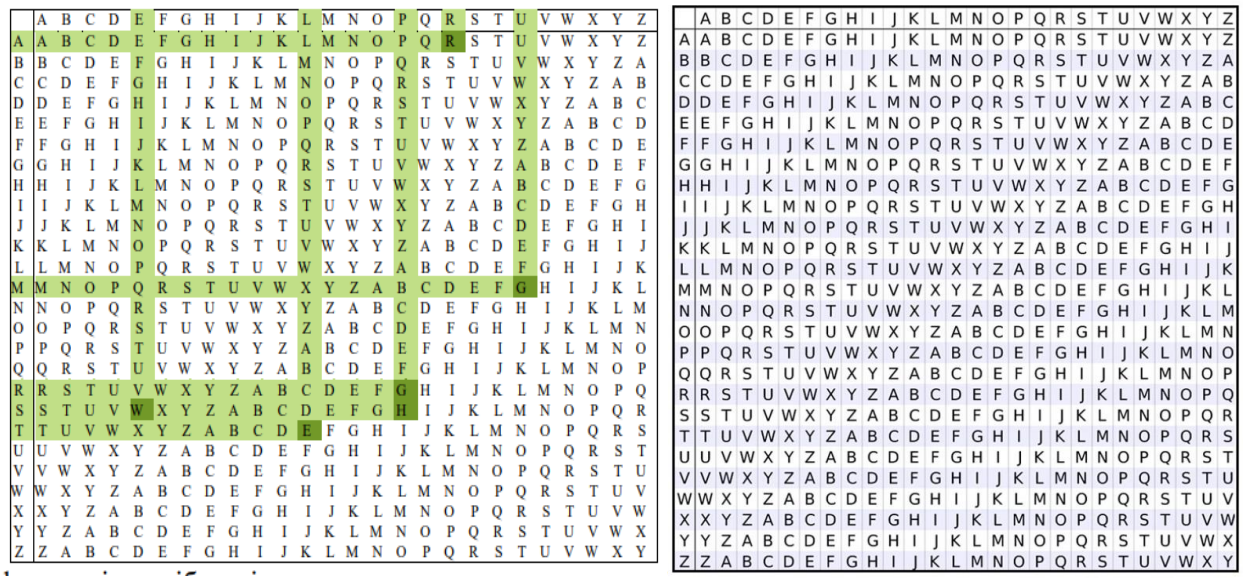


Рис. 1.9. Приклад шифрувальної таблиці Віженера.

Контрольні запитання.

1. Що таке криптографічний алгоритм та шифр?
2. Що таке криптографічний ключ?
3. Назвіть складові криптографічної системи.
4. У чому полягає криптостійкість криптографічної системи?
5. Опишіть алгоритм шифрування Цезаря.
6. У чому суть методу частотного криптоаналізу?
7. Опишіть алгоритм шифру Плейфера.
8. Що є ключем у шифрі Трітеміуса?

Завдання до лабораторної роботи.

I Зашифрувати своє прізвище ім‘я та по-батькові за допомогою зазначених алгоритмів (за необхідності перетворити ПІБ в транслітерований текст Сідоренко –Sidorenko):

- шифру Цезаря;
- Полібіанського квадрата;
- шифруючої системи Трисемуса;
- шифру Playfair;
- азбуки Морзе;
- системи омофонів (допускається для кожної літери алфавіту навести всього по дві шифрозаміни, тобто прийняти, що всі літери мають однакову ймовірність появи в текстах);
- шифру Віженера.

II Дешифрувати отримані в (I) зашифровані тексти.

При оформленні звіту необхідно навести вихідне повідомлення, таблицю шифрозамін, ключ (якщо таблиця шифрозамін не є ключем) та зашифроване повідомлення.

Лабораторна робота №2

Класичні алгоритми симетричного та асиметричного шифрування. Шифр Кардано. Шифри переставляння

Теоретичні відомості та приклади застосування.

Загальна характеристика шифрів перестановки (переставляння).

Шифр перестановки полягає в тому, що окремі знаки або певні групи знаків за певними правилами переставляються місцями безпосередньо у відкритому тексті.

У найпростішому випадку шифр перестановки використовується як блоковий. Це означає, що в процесі шифрування знаки відкритого тексту переставляють в межах деяких блоків фіксованого розміру.

Стійкість шифру перестановки залежить від розміру блоку, а також від рівня складності порядку перестановки.

Звичайна перестановка

Розглянемо блок відкритого тексту $T=(T_1,T_2,T_3,...,T_N)$ довжиною N і відповідний блок попарно різних індексів $\sigma=\sigma(K_1,K_2,K_3,...,K_N)$, де $1 \le K_i \le N$ для всіх $1 \le i \le N$. Тут блок індексів σ є ключем шифрування.

Звичайною перестановкою знаків даного тексту T називається його перевпорядкування таким чином, що знак з позиції $\sigma(i)=K_i$ у відкритому тексті переміщується у позицію і у криптограмі.

Приклад 2.1. Зашифруйте відкритий текст "ШИФРУВАННЯ ПЕРЕСТАНОВКОЮ". Для шифрування використайте ключ $\sigma = (3, 8, 1, 5, 2, 7, 6, 4)$ для блоку довжиною N=8.

	1	2	3	4	5	6	7	8
Текст	Ш	И	Ф	Р	У	В	А	Н
	3 – Ф	8 – Н	1 – Ш	5 – У	2 – И	7 – А	6 – В	4 – Р

В результаті маємо криптограму

"ФНШУИАВР_СНЕЯЕРПНЮТВАОКО".

Загальна можлива кількість перестановок заданого тексту T із N знаків рівна N!, значення якого швидко зростає зі збільшенням N.

Звичайні рядково-стовпчикові табличні перестановки

Це перестановки, коли даний текст записується у прямокутну таблицю певного розміру по рядках (послідовне заповнення її рядків зліва направо), а зашифрований текст отримують шляхом прочитування її послідовних стовпчиків згори вниз.

Приклад 2.2. Для повідомлення «ПЕРЕСТАНОВКИ» за допомогою шифрувальної таблиці розміром 3х4 буде отримано криптограму «псоетвракени».

п	е	р	е
с	т	а	н
о	в	к	и

Ключем шифру є розмір таблиці (для наведеного прикладу 3х4).

Для отримання вихідного повідомлення, слід вписати криптограму в таблицю того ж самого розміру по стовпчиках, а прочитати по рядках.

Таку перестановку називають *звичайною стовпчиково-рядковою.*

Рядково-стовпчикові табличні перестановки із застосуванням ключа стовпчиків

Процес шифрування полягає в тому, що над верхнім рядком таблиці записують ключ – вектор індексів перестановок стовпчиків. Після цього здійснюють вписування відкритого тексту у таблицю по рядках звичайним способом.

Приклад 2.3 Для повідомлення «ПЕРЕСТАНОВКИ» та ключа (4, 1, 3, 2) криптограма утворюється шляхом прочитування по стовпчиках, які беруться у порядку, визначеному ключем. Таким чином отримуємо криптограму «етвениракпсо».

4	1	3	2
п	е	р	е
с	т	а	н
о	в	к	и

Аналогічна процедура для шифрування з ключем рядків.

Шифр Кардано. Шифрування за допомогою трафаретів.

У 1550 італієць Джероламо Кардано дав описання шифру, що називається гратковим, або трафаретом, в якому секретне послання виявлялося приховано усередині довшого і абсолютно безневинно виглядаючого відкритого тексту. Секретне повідомлення можна було виявити, наклавши на відкритий текст аркуш пергаментного паперу з прорізами (трафарет). Слова, що з'являлися в прорізах, і складали секретне послання.

У шифрах Джероламо Кардано в якості ключа використовують маску (трафарет), розмір якої співпадає з розміром таблиці перестановки, де четверта частина комірок прорізана таким чином, що за чотири оберти на 90˚ вони покривають усю таблицю (матрицю). Тому такі шифри відносять до шифрів-трафаретів.

Квадратним трафаретом називають нанесену на планшет квадратну матрицю з прорізаними віконцями. Планшет, як маску, накладають на папір того ж розміру, і у віконця вписують знаки повідомлення по порядку слідування рядків зліва направо. Після першого заповнення планшет повертають на 90° за годинниковою (або проти годинникової стрілки) стрілкою і процедуру вписування повторюють. Таким способом вписування знаків повідомлення у віконця може бути здійснене чотири рази.

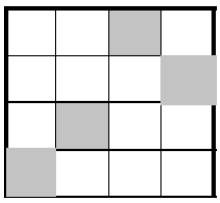
Квадратний трафарет цікавий тим, що після кожного повороту віконця знаходяться над незаповненими клітинками паперу. З цією метою розташування віконець на трафареті підбирають спеціально. Кількість віконець не повинна перевищувати четвертої частини від загальної кількості клітинок трафарету.

Щоб описати трафарет, застосовують позначення: нуль – віконце відсутнє, одиниця – віконце є. Тоді трафарет можна представити у вигляді сукупності двійкових чисел, кожне з яких відповідає його рядку. Ці числа можна представити у десятковій системі числення. Сукупність цих чисел являє собою ключ шифру. Якщо кількість віконець трафарету виявиться більшою кількості знаків повідомлення, то порожні віконця заповнюються випадковими знаками.

Для розшифрування криптограми потрібний такий же самий квадратний трафарет. В наведеному вище прикладі трафарет має чотири віконця, що дорівнює четвертій частині від загальної кількості клітинок 4*4=16. Слід відмітити, що для підвищення криптографічної стійкості кількість віконець трафарету можна зробити меншою.

Кількість можливих квадратних трафаретів різко зростає зі збільшенням їх розміру: один для 2*2, 256 для 4*4, більше 100 тисяч – для 6*6. Ця кількість суттєво збільшується за рахунок використання зменшення кількості віконець.

Приклад 2.4 Перетворити повідомлення «перестановки» за допомогою квадратного трафарета розміром 4х4, що має чотири віконця. Ключ (2, 1, 4, 8).



0□	0□	1□	0□
0□	0□	0□	1□
0□	1□	0□	0□
1□	0□	0□	0□

$0010_2=2_{10}$

$0001_2=1_{10}$

$0100_2=4_{10}$

$1000_2=8_{10}$

Ключем цього шифру є розмір квадратного трафарету 4*4 та послідовність двійкових чисел «0010, 0001, 0100, 1000» або, що те ж саме, десяткових чисел «2, 1, 4, 8», які цей квадратний трафарет описують.

		п		с							о		а		
			е		т					в		б			
	р						а	к						в	
Е						н			и						г

с	а	п	о
б	т	в	е
к	р	в	а
е	и	н	г

0° 90° 180° 270°

Рис. 2.1 Приклад застосування шифру Кардано для трафарету 4×4.

На останньому четвертому повороті трафарету знаки повідомлення вже вичерпались, і тому порожні віконця були заповнені додатковими випадковими знаками «абвг». Отримана криптограма «сапобтвекрваеинг».

Для дешифрування потрібний такий же самий квадратний трафарет.

Застосування магічних квадратів

Магічними квадратами називають квадратні таблиці, в кожну клітинку яких вписано послідовні натуральні числа починаючи з 1, які дають у сумі по

кожному стовпцю, кожному рядку і кожній діагоналі те саме число.

Існує тільки один магічний квадрат розміром 3×3 (якщо не враховувати його повороти). Кількість магічних квадратів 4×4 становить уже 880, а кількість магічних квадратів 5×5 – близько 250000.

Текст, що шифрується, вписується в магічні квадрати відповідно до нумерації їх клітинок. Якщо потім виписати вміст такої таблиці по рядках, то вийде шифротекст, сформований завдяки перестановці букв вихідного повідомлення.

Приклад 2.5 За допомогою магічного квадрату розміром 4×4 (рис. 2.2) виконайте шифрування тексту «ВІРТУАЛЬНИЙ КАНАЛ» .

16	3	2	13
5	10	11	8
9	6	7	12
4	15	14	1

Л	Р	І	А
У	И	Й	Ь
Н	А	Л	К
Т	А	Н	В

Рис. 2.2 Приклад магічного квадрата 4×4 та його заповнення

Шифротекст, який одержали при зчитуванні вмісту правої таблиці по рядках, має вигляд: ЛРІА УИЙЬ НАЛК ТАНВ або ЛРІАУИЙЬНАЛКТАНВ.

Контрольні запитання.

1. Що таке криптографічний алгоритм та шифр Кардано?
2. Що є криптографічним ключем у шифрах, що використовують рядково-стовпчикові табличні перестановки?
3. Назвіть складові криптографічної системи заснованої на використанні магічних квадратів.
4. У чому полягає криптостійкість криптографічної системи?
5. Опишіть наведені в лабораторній роботі алгоритми шифрування переставлянням.

Завдання до лабораторної роботи.

I Зашифрувати своє прізвище ім‘я та по-батькові за допомогою зазначених алгоритмів (за необхідності перетворити ПІБ в транслітерований текст Сідоренко –Sidorenko):

- звичайної перестановки (ключ сформувати самостійно);
- магічного квадрату 3×3, та 4×4 (квадрат обрати самостійно та навести у відповіді);
- трафарету;
- рядково-стовпчикової табличної перестановки із застосуванням ключа стовпчиків (ключ обрати самостійно);

II Дешифрувати отримані в (I) зашифровані тексти.

При оформленні звіту необхідно навести вихідне повідомлення, таблицю шифрозамін, ключ (якщо таблиця шифрозамін не є ключем) та зашифроване повідомлення.

Лабораторна робота №3

Класичні алгоритми симетричного та асиметричного шифрування. Шифри гамування.

Теоретичні відомості та приклади застосування.

Шифрування методом складної заміни

Шифрування вхідної інформації методом складної заміни називають багатоалфавітним або поліалфавітним, оскільки для шифрування кожного символу початкового алфавіту застосовують свій шифр складної заміни.

Поліалфавітні шифри заміни запропонував і увів у практику криптографії Леон Батист Альберті в 1566 р.

Поліалфавітне підставляння послідовно і циклічно міняє алфавіти, що використовуються.

При r-алфавітній підстановці символ x0 початкового повідомлення замінюється символом y0 з алфавіту B0, символ x1 – символом y1 з алфавіту B1, і так далі, символ xr-1 замінюється символом yr-1 з алфавіту Br-1, символ xr замінюється символом yr знову з алфавіту B0 і так далі.

Загальна схема поліалфавітного підставляння для випадку r = 4 показана на рис. 3.1.

Вхідний символ:	x_0	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9
Алфавіт підставляння:	B_0	B_1	B_2	B_3	B_0	B_1	B_2	B_3	B_0	B_1

Рис. 3.1. Схема r-алфавітного підставляння для випадку r = 4

Ефект використання поліалфавітного підставляння полягає в тому, що забезпечується маскування природної статистики початкової мови, оскільки конкретний символ з початкового алфавіту A може бути перетворений в декілька різних символів шифрувальних алфавітів Bj, $j = \overline{1,r}$. Ступінь забезпечуваного захисту теоретично пропорційний довжині періоду r в послідовності використовуваних алфавітів Bj.

Шифр Гронсфельда

Шифр Гронсфельда, був запропонований Хосе де Бронкхорстом (графом Гронсфельдом) близько 1744 року.

Шифр Гронсфельда, є модифікацією шифру Віженера з числовим ключем. Для цього під буквами початкового повідомлення записують цифри числового ключа. Якщо ключ коротший за повідомлення, то його запис циклічно повторюють.

Зашифрований текст отримують за наступним правилом: заміна для кожної букви визначається як у шифрі Цезаря з локальним ключем, що дорівнює числу ключа, який стоїть в цьому стовпчику.

Приклад 3.1 Зашифруйте шифром Гронсфельда з ключем «123412341» текст - "GRONSFELD".

Складаємо таблицю замін. а клавіша 1234 починається зі зміщення G на 1 позицію в алфавіті, стає H, потім R зміщується на 2 позиції і стає T тощо. Зашифрований текст: HTRRTHHPE

Повідомлення	G	R	O	N	S	F	E	L	D
Ключ	1	2	3	4	1	2	3	4	1
Зашифрований текст									

Повідомлення	G	R	O	N	S	F	E	L	D
Ключ	1	2	3	4	1	2	3	4	1
Зашифрований текст	H	T	R	R	T	H	H	P	E

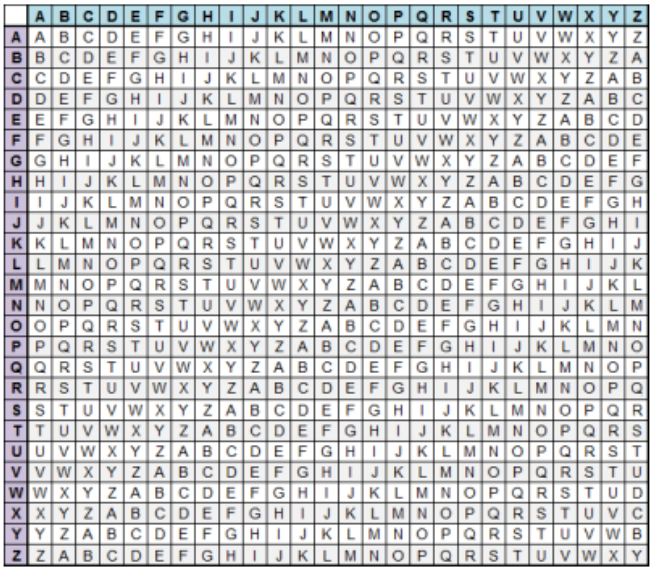


Рис. 3.2. Приклад застосування шифру Гронсфельда з використанням таблиці Віженера.

Гамування

Найпростішою і в той же час найбільш надійною з усіх схем шифрування є так звана схема одноразового (однократного) використання (рис. 1), винахід, який найчастіше пов'язують з ім'ям Г.С. Вернама.

Гама шифру - псевдовипадкова послідовність, створена згідно з заданим алгоритмом для зашифрування відкритих даних і розшифрування зашифрованих даних.

Гамування – це накладання за певним законом на відкриті дані (дані, що підлягають зашифровуванню) криптографічної гами, тобто, послідовності елементів даних, що виробляються за допомогою деякого криптографічного алгоритму, для отримання зашифрованих даних.

Процес зашифрування полягає в генерації гами шифру і накладанні отриманої гами на вихідний відкритий текст визначеним чином, наприклад, з використанням операції додавання за модулем 2.

Слід зазначити, що перед зашифруванням відкриті дані M розбивають на блоки M^i однакової довжини, як правило, по 32, 64, 128, 256 бітів. Гама шифру генерується у вигляді послідовності блоків G^i аналогічної довжини.

Рівняння зашифрування можна записати у вигляді $C^i = G^i \oplus M^i$, де $1 \le i \le m$, m – кількість блоків відкритого тексту; C^i – i-й блок шифротексту; G^i – i-й блок гами шифру; M^i – i-й блок відкритого тексту.

Процес розшифрування зводиться до повторного накладання тієї ж, або нової гами на зашифровані дані. Рівняння розшифрування має вигляд $M^i = G^i \oplus C^i$.

Шифр звичайного накладання двійкової гами

Продемонструємо на прикладі процес шифрування звичайним накладанням двійкової гами .

Приклад 3.1 Зашифруйте методом накладання гами текст «БАГАЖА», записаний в алфавіті обсягом m=8 знаків «АБВГДЖЗЕ». Кожний символ тексту та гами представити 3-розрядними двійковими числами.

1. Відкритий текст подають у вигляді неперервної послідовності k-розрядних двійкових чисел. Для цього використовують перетворення знаків відкритого тексту із алфавіту обсягом $m=2^k$.

Представимо букви алфавіту у вигляді 3-розрядних двійкових чисел:

Номер	0	1	2	3	4	5	6	7
Знак відкритого тексту	А	Б	В	Г	Д	Ж	З	Е
Двійковий код	000	001	010	011	100	101	110	111

2. Генерують гаму шифру у вигляді послідовності псевдовипадкових двійкових цифр.

Довжина гами повинна дорівнювати довжині тексту – в нашому випадку 6 букв по три розряди дають 18 розрядів.

Двійкову гаму можна придумати довільно (наприклад 010001011100111101) або, наприклад взяти перші літери довільного тексту та взяти їх двійкові коди із таблиці ASCII (наприклад А -0100 0001, N-0100 1110, W- 0101 0111, тоді гама –0100 00010100 111001).

Також поширеним методом генерування гами є використання певних правил, співвідношень, зокрема математичних, алгоритмів.

Наприклад двійкову гамму шифру можемо згенерувати лінійним методом, використовуючи теорію чисел – порівняння за модулем $x_{n+1}=(ax_n+b) \bmod m$, *n =0, 1, 2,… з обраними параметрами. Візьмемо, наприклад m=16, a=5, b=7,* $x_0=10$.

Отримаємо послідовність псевдовипадкових чисел: 10, 9, 4, 11, 14, …. Кожне із отриманих чисел представимо як чотирирозрядне двійкове $10=1010_2$, $9=1001_2$, $4=0100_2$, $11=1011_2$, $14=1110_2$. *Випишемо послідовність отриманих представлень 10101001010010111110, залишивши 18 розрядів*

3. На кожний черговий розряд відкритого тексту накладають відповідний розряд двійкової гами з використанням операції додавання по модулю 2 (виключна диз'юнкція, або операція XOR, додавання за модулем два – двомісна логічна операція, що приймає значення «істина» тоді і тільки тоді коли значення «істина» має рівно один з її операндів)і таким шляхом отримують черговий двійковий розряд криптограми.

Відкритий текст «БАГАЖА»	001|000|011|000|101|000
Гамма шифру	101|010|010|100|101|111
Результат додавання по модулю 2	100|010|001|100|000|111
Криптограма	Д В Б Д А Е

4. Подають криптограму через даний алфавіт, виконавши попереднє розбиття криптограми на послідовні к-розрядні двійкові числа.

Ключові дані, що входять до складу шифру, такі:

– алфавіт з порядковими номерами знаків, а також кількість розрядів двійкового їх подання;

– параметри обраного методу генерації двійкової гами шифру, в тому числі кількість розрядів двійкового подання кожного псевдовипадкового числа їх послідовності.

Розглянемо приклад. Для ілюстрації шифрування тексту методом гамування візьмемо двійкові коди символів відкритого тексту *M* та як гаму *G* – випадкову послідовність двійкових кодів.

Генерація гами за допомогою скремблера.

Скремблером називається програмна чи апаратна реалізація алгоритму, що дозволяє шифрувати побитно неперервні потоки інформації.

Розглянемо зсувний регістр зі зворотним зв'язком (LFSR – Linear Feedback Shift Register) - логічний пристрій, схема на рис.2.

Зсувний регістр є послідовністю *n* біт (*n*-бітний зсувний регістр). Щоразу, коли потрібно отримати біт, всі біти зсувного регістру пересуваються вправо на 1 позицію. Новий крайній лівий біт є функцією всіх інших бітів реєстру. На виході зсувного регістру виявляється молодший біт.

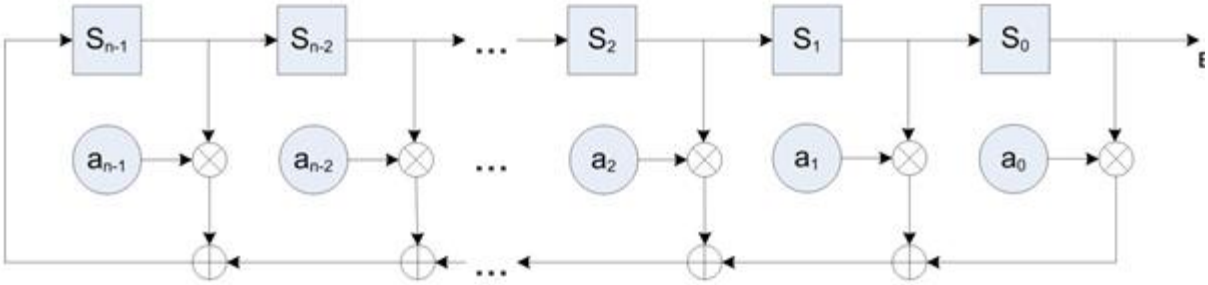


Рис. 3.3. Схема LFSR – Linear Feedback Shift Register

LFSR складається з осередків пам'яті, двійкові стани яких у моменти часу t=0,1,2,… характеризуються значеннями $S_0(t), S_1(t), \dots, S_{n-1}(t) \in A = \{0, 1\}$.. Виходи комірок пам'яті пов'язані не тільки послідовно один з одним, але і з суматорами $\oplus$ відповідно до коефіцієнтів передачі $a_0, a_1, \dots, a_{n-1} \in A$: якщо $a_i = 1$, то значення $S_i(t)$ i-ї комірки передається на один із входів і-го суматора; якщо ж $a_i = 0$, то така передача відсутня. Зазвичай коефіцієнти передачі задаються за допомогою полінома:

$$f(x) = x^n + a_{n-1} \cdot x^{n-1} + a_{n-2} \cdot x^{n-2} + \cdots + a_2 \cdot x^2 + a_1 \cdot x + a_0.$$

Стан LFSR в даний час задається двійковим n-вектор-стовпцем $S(t) = \left(S_{n-1}(t), \dots, S_0(t)\right)'$. Вміст комірок LFSR з часом змінюється так, визначаючи цим динаміку станів LFSR:

$$S_i(t+1) = \begin{cases} S_{i+1}(t), & i \in \overline{0, n-2}, \\ \sum_{j=0}^{n-1} a_j S_j(t), & i = n-1. \end{cases}$$

При досить довгій роботі скремблера неминуче виникає його зациклювання. Після виконання певного числа тактів в комірках скремблера створиться комбінація біт, яка в ньому вже одного разу виявлялася, і з цього моменту шифруюча послідовність почне циклічно повторюватися з фіксованим періодом.

Щоб побудувати N-розрядний скремблер, що створює послідовність найбільшої довжини, користуються примітивними многочленами. Примітивний (базовий) многочлен ступеня по модулю 2 – це незвідний многочлен, який є дільником $x^{2^n-1}+1$, але не є дільником x^d+1 для всіх d, на які ділиться 2^n-1. Незведений многочлен ступеня *n* не можна у вигляді добутку інших многочленів, крім нього самого та одиничного.

Знайдений примітивний многочлен ступеня *n* записується в двійковому вигляді, потім відкидається одиниця, що відповідає найстаршому розряду.

Наведемо приклад 7-розрядного скремблера, що генерує послідовність з рівним періодом : $T = 7$: $x^7 + x^6 + x^2$. Нехай початкове значення стану дорівнюватиме $(1001111)_2$.

Для цього зсувного регістру новий біт генерується за такою схемою:

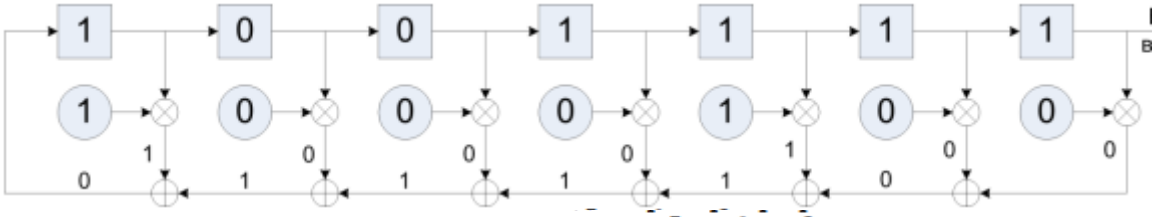


Рис. 3.4 Схема LFSR для многочлена $T = 7$: $x^7 + x^6 + x^2$ ***при початковому стані*** $(1001111)_2$.

Послідовність зміни внутрішнього стану скремблер має вигляд:
1001111→0100111→1010011→1101001→1110100→0111010→0011101→1001110→0100111→1010011 ...

Завдання до лабораторної роботи

І. Реалізувати програму для шифрування, що дозволяє виконувати наступні дії:

1. Шифрувати дані в режимі одноразового гамування:

1) текст, що шифрується, повинен зберігатися у файлі;

2) ключ шифрування повинен задаватися випадковим чином;

3) зашифрований текст повинен зберігатися в один файл, а ключ, що використовувався при шифруванні, - в інший;

4) у процесі шифрування передбачити можливість перегляду та зміни ключа, шифрованого та зашифрованого текстів у двійковому та символьному вигляді.

2. Шифрувати дані за допомогою кожного заданого у варіанті скремблера:

1) текст, що шифрується, повинен зберігатися в одному файлі, початкове значення скремблера - в іншому;

2) зашифрований текст повинен зберігатися у файл;

II. Реалізувати програму для дешифрування, що дозволяє виконувати такі дії:

1.Дешифрувати дані в режимі одноразового гамування:

1) зашифрований текст повинен зберігатися в одному файлі, ключ – в іншому;

2) розшифрований текст повинен зберігатись у файл;

2. Розшифрувати дані за допомогою кожного заданого у варіанті скремблера:

1) зашифрований текст повинен зберігатися в одному файлі, початкове значення скремблера – в іншому;

2) зашифрований текст повинен зберігатися у файл;

III. Протестувати правильність роботи розроблених програм на наступних задачах:

1) M = «*НІЖНО ВПЛІТАЄТЬСЯ В ГОМІН ДНІПРА ДОБРЕ І ЩИРЕ ШЕВЧЕНКІВСЬЕ СЛОВО*»

Key=«*СКОМАРОВСЬКИЙ*».

2) Виконати розшифрування шифротексту *C* (ключове слово – «EMPIRE»). При розшифруванні врахувати, що алфавіт містить пробіл, за яким ідують символи латинського алфавіту

C = «MRPPI FGOUM RYMAH NRYMD UNRWZ OANJE FTIZNI MIQWR EQUNG EIALW RNSMX RJEUA STWYE NCMRY MRCIZ JEEIC XKJQP QXEBD LBJYM LRKME TGJJX EEDIK MFFPB ZJEZD WWCEI DCCIE DBRKF YAIFZ Y».

№	**Скремблер**	№	**Скремблер**
1	$x^9 + x^4 + 1$		$x^8 + x^4 + x^3 + x^2 + 1$
2	$x^9 + x^3 + 1$		$x^8 + x^5 + x^3 + x^2 + 1$
3	$x^{10} + x^3 + 1$		$x^8 + x^4 + x^3 + x^2 + 1$
4	$x^{10} + x^7 + 1$		$x^8 + x^6 + x^2 + 1$
5	$x^{11} + x^2 + 1$		$x^7 + x^5 + x^2 + 1$
6	$x^{11} + x^5 + x^2 + 1$		$x^7 + x + 1$
7	$x^{11} + x^2 + 1$		$x^6 + x + 1$
8	$x^{11} + x^3 + x^2 + 1$		$x^6 + x^5 + x + 1$
9	$x^{12} + x^6 + x^4 + x + 1$		$x^5 + x^2 + 1$
10	$x^{12} + 1$		$x^5 + x^4 + x + 1$

Лабораторна робота №4

Класичні алгоритми симетричного та асиметричного шифрування. Мережа Фейстеля.

Теоретичні відомості та приклади застосування.

Мережа Фейстеля

У 1971 році Хорст Фейстель *(Horst Feistel)* запатентував два пристрої для реалізації різних алгоритмів шифрування, які отримали назву "Люцифер" *(Lucifer).* Один із пристроїв використовував конструкцію, згодом названу "мережею Фейстеля" *(Feistel cipher, Feistel network).* Зауважимо, що проєкт "Люцифер" став базисом для алгоритму *DES (Data Encryption Standard).* У 1973 році Хорст Фейстель навів опис першої версії проєкту "Люцифер".

Опис алгоритму. Мережа Фейстеля здійснює розбиття оброблюваного блока даних на кілька підблоків (найчастіше - на два), один із яких обробляють деякою функцією *f* (a), а потім накладають на один або кілька інших підблоків. На рис. 4.1 наведено найпоширенішу структуру алгоритмів на основі мережі Фейстеля.

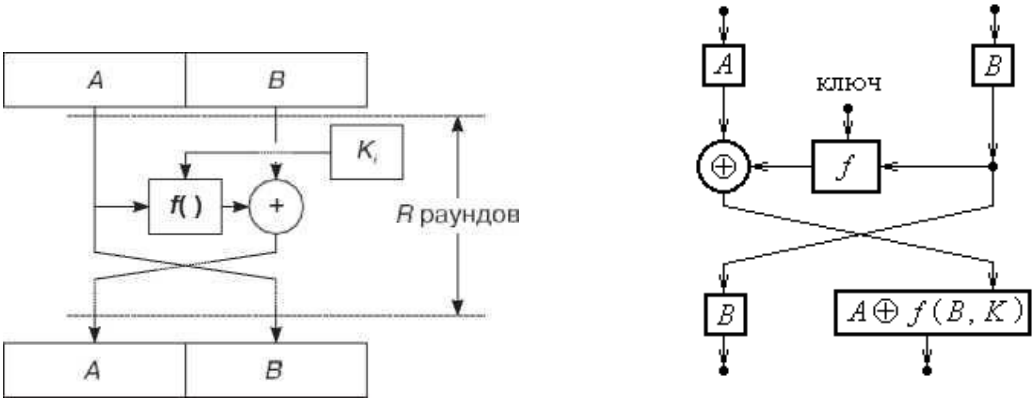


Рис. 4.1 **Структура алгоритмів на основі мережі Фейстеля**

Блок відкритого тексту ділиться на дві рівні частини *A* - ліва (L) і *B* - права *(R),* у кожному раунді i (i = 1 ... *n* - номер раунду) обчислюється

$$L_i = R_{i-1} \oplus f(L_{i-1}, K_{i-1});$$
$$R_{i-1} = L_{i-1},$$

де *f* () - деяка функція, а K_{i-1} - ключ i-го раунду.

Результатом виконання *n* раундів є *(Ln , Rn*), але зазвичай у n-му раунді перестановка *Ln* і *Rn* не здійснюється, що дає змогу використовувати ту саму процедуру і для розшифрування, просто інвертувавши порядок використання раундової ключової інформації:

$$L_{i-1} = R_i \oplus f(L_i, K_{i-1});$$
$$R_{i-1} = L_i.$$

Одна з переваг такої моделі - оборотність алгоритму незалежно від використовуваної функції f причому вона може бути як завгодно складною.

Надійність алгоритму. Мережі Фейстеля були широко вивчені криптографами через їхнє широке поширення. У 1988 році Майкл Любі *(Michael Luby)* і Чарльз Ракофф *(Charles Rackoff)* провели дослідження мережі Фейстеля та довели, що якщо раундова функція є криптостійкою псевдовипадковою, а ключі, які використовують, незалежні в кожному раунді, то трьох раундів буде достатньо для того, щоб блоковий шифр був псевдовипадковою перестановкою.

У багатьох блокових шифрах на основі мережі Фейстеля було знайдено ті чи інші вразливості, однак у низці випадків ці вразливості є суто теоретичними і за нинішньої продуктивності комп'ютерів використовувати їх на практиці для злому неможливо.

Приклад 4.1. В якості засобу шифрування обрана 2-раундова мережа Фейстеля. F -функція задана таблицею 4.1.

Таблиця 4.1 F -функції

аргументи	1	2	3	4
0000	0111	1011	1100	1011
0001	0100	0110	0111	0100
0010	1100	1010	1111	0011
0011	1101	0111	1010	1010
0100	0101	0011	1000	1100
0101	0011	1000	1010	1111
0110	1000	0010	0100	1001
0111	1011	0111	1010	0111
1000	0101	1010	0011	1110
1001	0011	0100	0111	0110
1010	1011	1111	0110	1110
1011	1100	0110	1000	1100
1100	1011	1001	0011	0111
1101	0011	1100	1111	0011
1110	0100	1111	1110	1000
1111	1000	1011	1010	0001

Зашифруємо за допомогою мережі літеру «И» з ключами K_2, K_4. Для цього представимо літеру в виді двійкової послідовності: 11001000. Процес і результат шифрування наведено на рис. 4.2. Отже, шифротекст має вид: 00010110.

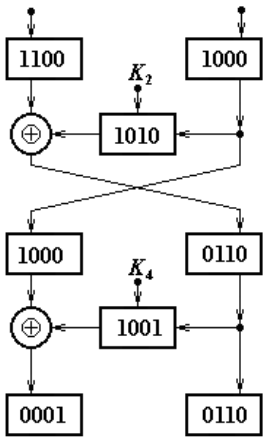


Рис. 4.2. Шифрування мережею Фейстеля з двома раундами.

Контрольні запитання

1. У чому полягає алгоритм одноразового блокноту?
2. Що являє собою операція XOR?
3. Які переваги і недоліки шифрування методом одноразового блокноту?
4. До яких шифрів належить стандарт шифрування даних DES?
5. Якою повинна бути довжина ключа у мережі Фейстеля?
6. З яких кроків складається алгоритм мережі Фейстеля.
7. Скільки раундів має мережа Фейстеля?

Завдання на проведення лабораторної роботи.

1. Зашифруйте Ваше прізвище мережею Фейстеля з двома раундами. В якості ключів раундів обрати імя та по батькові.

2. Розшифрувати символ за допомогою мережі Фейстеля. Варіанти завдань і ключі наведені в таблиці 4.2.

Таблиця 4.2 − Варіанти завдань

№	шифротекст	ключі
1	00110000	4,1
2	10110100	4,1
3	00001011	4,1
4	01100101	2,3
5	11000110	2,3
6	01000110	2,3
7	01010001	1,2
8	01001000	1,2
9	00001001	1,2
10	10010110	2,4
11	11011011	2,4
12	01110100	3,2
13	01000111	3,2
14	01111011	3,2
15	00001011	4,4
16	01110111	4,4
17	10010010	4,4
18	01110000	1,3
19	11111111	1,3
20	01101001	1,3

3. За допомогою навчальної програми CrypTool 2 – портал безкоштовних програм електронного навчання в галузі криптографії та криптоаналізу The CrypTool Portal. URL: http://www.cryptool.org/en перевірити отримані результати п.1, 2.