**Міністерство освіти та науки України**

**НТУУ «КПІ ім. І.Сікорського»**

**Кафедра АПЕПС**

Звіт до лабораторної роботи №8

з предмету: «Безпека програм і даних»

на тему: «Система шифрування des»

12 варіант

Виконав: студент 4 курсу  
 ТЕФ, ТІ-31  
 Ніколаєв В.В.

Перевірив: Тарнавський Ю.А.

Київ-2016

Цілі роботи:

Реалізувати шифрування des в прикладній програмі

Data Encryption Standart

|  |  |
| --- | --- |
| **Алгоритм блочного шифрування** | |
| [Data Encription Standard Flow Diagram.svg](https://uk.wikipedia.org/wiki/%D0%A4%D0%B0%D0%B9%D0%BB:Data_Encription_Standard_Flow_Diagram.svg) | |
| **Назва:** | DES (Data Encryption Standard), DEA (Data Encryption Algorithm) |
| **Розробник:** | [IBM](https://uk.wikipedia.org/wiki/IBM) |
| **Створений:** | [1975](https://uk.wikipedia.org/wiki/1975) р. |
| **Опублікований:** | [1975](https://uk.wikipedia.org/wiki/1975) р. |
| **Розмір ключа:** | 56 [біт](https://uk.wikipedia.org/wiki/%D0%91%D1%96%D1%82" \o "Біт) |
| **Розмір блоку:** | 64 [біт](https://uk.wikipedia.org/wiki/%D0%91%D1%96%D1%82" \o "Біт) |
| **Число раундів:** | 16 |
| **Тип:** | [Мережа Фейстеля](https://uk.wikipedia.org/wiki/%D0%9C%D0%B5%D1%80%D0%B5%D0%B6%D0%B0_%D0%A4%D0%B5%D0%B9%D1%81%D1%82%D0%B5%D0%BB%D1%8F) |

**DES** ([англ.](https://uk.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D1%96%D0%B9%D1%81%D1%8C%D0%BA%D0%B0_%D0%BC%D0%BE%D0%B2%D0%B0" \o "Англійська мова) *Data Encryption Standard*) — це симетричний алгоритм шифрування певних даних , стандарт шифрування прийнятий урядом [США](https://uk.wikipedia.org/wiki/%D0%A1%D0%A8%D0%90) із 1976 до кінця 1990-х, з часом набув міжнародного застосування. Ще з часу свого розроблення алгоритм викликав неоднозначні відгуки. Оскільки DES містив засекречені елементи своєї структури, породжувались побоювання щодо можливості контролю з боку Національного Агентства Безпеки США ([англ.](https://uk.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D1%96%D0%B9%D1%81%D1%8C%D0%BA%D0%B0_%D0%BC%D0%BE%D0%B2%D0%B0" \o "Англійська мова) *National Security Agency*). Алгоритм піддавався критиці за малу довжину ключа, що, врешті, після бурхливих обговорень та контролю академічної громадськості, не завадило йому стати загальноприйнятим стандартом. DES дав поштовх сучасним уявленням про [блочні алгоритми шифрування](https://uk.wikipedia.org/wiki/%D0%91%D0%BB%D0%BE%D1%87%D0%BD%D1%96_%D0%B0%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC%D0%B8_%D1%88%D0%B8%D1%84%D1%80%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F) та [криптоаналіз](https://uk.wikipedia.org/wiki/%D0%9A%D1%80%D0%B8%D0%BF%D1%82%D0%BE%D0%B0%D0%BD%D0%B0%D0%BB%D1%96%D0%B7" \o "Криптоаналіз).

Зараз DES вважається ненадійним в оcновному через малу довжину ключа (56 біт) та розмір блоку (64 біти). У [1999](https://uk.wikipedia.org/wiki/1999) ключ DES було публічно дешифровано за 22 години 15 хвилин. Вважається, що алгоритм достатньо надійний для застосування у модифікації [3-DES](https://uk.wikipedia.org/wiki/3-DES), хоча існують розроблені теоретичні атаки. DES поступово витісняється алгоритмом [AES](https://uk.wikipedia.org/wiki/Advanced_Encryption_Standard), що з 2002 року є стандартом США.

Строго кажучи, існує різниця між стандартом DES (Data Encryption Standard) та алгоритмом DEA (Data Encryption Algorithm).

Історія DES

Історія розробки DES сягає початку 1970-х і почалась за ініціативи Національного Бюро Стандартів [США](https://uk.wikipedia.org/wiki/%D0%A1%D0%A8%D0%90) ([англ.](https://uk.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D1%96%D0%B9%D1%81%D1%8C%D0%BA%D0%B0_%D0%BC%D0%BE%D0%B2%D0%B0" \o "Англійська мова) *National Bureau of Standards*) — тепер, Національний Інститут Стандартів і Технологій ([англ.](https://uk.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D1%96%D0%B9%D1%81%D1%8C%D0%BA%D0%B0_%D0%BC%D0%BE%D0%B2%D0%B0" \o "Англійська мова) *National Institute of Standards and Technology*) — для забезпечення засекречених урядових даних. [15 травня](https://uk.wikipedia.org/wiki/15_%D1%82%D1%80%D0%B0%D0%B2%D0%BD%D1%8F) [1973](https://uk.wikipedia.org/wiki/1973), після консультування із Національним Бюром Безпеки США, НБС оголосило конкурс на розробку алгоритму шифрування який би відповідав поставленим строгим архітектурним вимогам, однак, таких пропозицій не надійшло. Лише на другий конкурс [27 серпня](https://uk.wikipedia.org/wiki/27_%D1%81%D0%B5%D1%80%D0%BF%D0%BD%D1%8F) [1974](https://uk.wikipedia.org/wiki/1974) [IBM](https://uk.wikipedia.org/wiki/IBM) подала розробку яка задовільняла поставленим вимогам — алгоритм, розроблений в період 1973–1974, в основу якого був покладений шифр [Хорста Фейстеля](https://uk.wikipedia.org/wiki/%D0%A5%D0%BE%D1%80%D1%81%D1%82_%D0%A4%D0%B5%D0%B9%D1%81%D1%82%D0%B5%D0%BB%D1%8C" \o "Хорст Фейстель) [Люцифер](https://uk.wikipedia.org/w/index.php?title=%D0%9B%D1%8E%D1%86%D0%B8%D1%84%D0%B5%D1%80_(%D1%88%D0%B8%D1%84%D1%80)&action=edit&redlink=1).

**Хронологія**

|  |  |  |
| --- | --- | --- |
| **Дата** | **Рік** | **Подія** |
| [15 травня](https://uk.wikipedia.org/wiki/15_%D1%82%D1%80%D0%B0%D0%B2%D0%BD%D1%8F) | 1973 | Національне Бюро Стандартів (НБС) США опублікувало перший запит на стандарт алгоритму шифрування. Ніхто не відреагував |
| [27 серпня](https://uk.wikipedia.org/wiki/27_%D1%81%D0%B5%D1%80%D0%BF%D0%BD%D1%8F) | 1974 | НБС опублікувало другий запит на алгоритм шифрування. [IBM](https://uk.wikipedia.org/wiki/IBM) подала заявку |
| [17 березня](https://uk.wikipedia.org/wiki/17_%D0%B1%D0%B5%D1%80%D0%B5%D0%B7%D0%BD%D1%8F) | 1975 | DES опубліковано у [*Federal Register*](https://uk.wikipedia.org/wiki/Federal_Register) для коментарів |
| Серпень | 1976 | Перший робоча зустріч по DES |
| Вересень | 1976 | Друга робоча група, обговорення математичного підґрунтя DES |
| Листопад | 1976 | DES погоджено як стандарт |
| [15 січня](https://uk.wikipedia.org/wiki/15_%D1%81%D1%96%D1%87%D0%BD%D1%8F) | 1977 | DES опубліковано як стандарт FIPS PUB 46 |
|  | 1983 | DES знову підтверджується як стандарт |
| [22 січня](https://uk.wikipedia.org/wiki/22_%D1%81%D1%96%D1%87%D0%BD%D1%8F) | 1988 | DES вдруге підтверджується як стандарт (FIPS 46-1) |
| Липень | 1990 | Біхем і Шамір заново відкривають [диференціальний криптоаналіз](https://uk.wikipedia.org/wiki/%D0%94%D0%B8%D1%84%D0%B5%D1%80%D0%B5%D0%BD%D1%86%D1%96%D0%B0%D0%BB%D1%8C%D0%BD%D0%B8%D0%B9_%D0%BA%D1%80%D0%B8%D0%BF%D1%82%D0%BE%D0%B0%D0%BD%D0%B0%D0%BB%D1%96%D0%B7" \o "Диференціальний криптоаналіз), і застосовують його до 15-раундової DES-подібної криптосистеми. |
|  | 1992 | Біхем і Шамір описують теоретичну атаку із меншою часовою складністю ніж повний перебір: [диференціальний криптоаналіз](https://uk.wikipedia.org/wiki/%D0%94%D0%B8%D1%84%D0%B5%D1%80%D0%B5%D0%BD%D1%86%D1%96%D0%B0%D0%BB%D1%8C%D0%BD%D0%B8%D0%B9_%D0%BA%D1%80%D0%B8%D0%BF%D1%82%D0%BE%D0%B0%D0%BD%D0%B0%D0%BB%D1%96%D0%B7" \o "Диференціальний криптоаналіз). Однак, для її успішності необхідно 247незашифрованих даних. |
| [30 грудня](https://uk.wikipedia.org/wiki/30_%D0%B3%D1%80%D1%83%D0%B4%D0%BD%D1%8F) | 1993 | DES втретє підтверджується як стандарт (FIPS 46-2) |
|  | 1994 | Вперше проведено експериментальний криптоаналіз DES використовуючи [лінійний криптоаналіз](https://uk.wikipedia.org/w/index.php?title=%D0%9B%D1%96%D0%BD%D1%96%D0%B9%D0%BD%D0%B8%D0%B9_%D0%BA%D1%80%D0%B8%D0%BF%D1%82%D0%BE%D0%B0%D0%BD%D0%B0%D0%BB%D1%96%D0%B7&action=edit&redlink=1" \o "Лінійний криптоаналіз (ще не написана)) (Matsui, 1994). |
| червень | 1997 | Вперше публічно зломано зашифроване повідомлення DES. Проект DESCHALL |
| липень | 1998 | [EFF](https://uk.wikipedia.org/wiki/Electronic_Frontier_Foundation)'s [DES cracker](https://uk.wikipedia.org/w/index.php?title=EFF_DES_cracker&action=edit&redlink=1) (Deep Crack) дешифрує ключ DES за 56 годин. |
| січень | 1999 | Разом [Deep Crack](https://uk.wikipedia.org/w/index.php?title=Deep_Crack&action=edit&redlink=1) та [distributed.net](https://uk.wikipedia.org/w/index.php?title=Distributed.net&action=edit&redlink=1) дешифрують ключ DES за 22 години 15 хвилин. |
| [25 жовтня](https://uk.wikipedia.org/wiki/25_%D0%B6%D0%BE%D0%B2%D1%82%D0%BD%D1%8F) | 1999 | DES вчетверте підтверджується як стандарт (FIPS 46-3), в якому зазначено, що надавати перевагу слід [3-DES](https://uk.wikipedia.org/wiki/3-DES). |
| [26 листопада](https://uk.wikipedia.org/wiki/26_%D0%BB%D0%B8%D1%81%D1%82%D0%BE%D0%BF%D0%B0%D0%B4%D0%B0) | 2001 | [AES](https://uk.wikipedia.org/wiki/Advanced_Encryption_Standard) опубліковано як стандарт (FIPS 197) |
| [26 травня](https://uk.wikipedia.org/wiki/26_%D1%82%D1%80%D0%B0%D0%B2%D0%BD%D1%8F) | 2002 | AES вступає в силу |

Опис алгоритму

DES є блочним [шифром](https://uk.wikipedia.org/wiki/%D0%A8%D0%B8%D1%84%D1%80) - дані шифруються [блоками](https://uk.wikipedia.org/wiki/%D0%91%D0%BB%D0%BE%D0%BA) по 64 біти - 64 бітний блок явного тексту подається на вхід алгоритму, а 64-бітний блок шифрограми отримується в результаті роботи алгоритму. Крім того, як під час [шифрування](https://uk.wikipedia.org/wiki/%D0%A8%D0%B8%D1%84%D1%80%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F), так і під час дешифрування використовується один і той самий алгоритм (за винятком дещо іншого шляху утворення робочих ключів).

Ключ має довжину 56 [біт](https://uk.wikipedia.org/wiki/%D0%91%D1%96%D1%82) (як правило, в джерельному вигляді ключ має довжину 64 біти, де кожний 8-й біт є бітом паритету, крім того, ці контрольні біти можуть бути винесені в останній байт ключа). Ключем може бути довільна 64-бітна [комбінація](https://uk.wikipedia.org/w/index.php?title=%D0%9A%D0%BE%D0%BC%D0%B1%D1%96%D0%BD%D0%B0%D1%86%D1%96%D1%8F&action=edit&redlink=1), яка може бути змінена у будь-який момент часу. Частина цих комбінацій вважається слабкими [ключами](https://uk.wikipedia.org/wiki/%D0%9A%D0%BB%D1%8E%D1%87), оскільки може бути легко визначена. Безпечність алгоритму базується на безпечності ключа.

На найнижчому рівні алгоритм є ніщо інше, ніж поєднання двох базовних технік шифрування: [перемішування](https://uk.wikipedia.org/wiki/%D0%9F%D0%B5%D1%80%D0%B5%D1%81%D1%82%D0%B0%D0%BD%D0%BE%D0%B2%D0%BE%D1%87%D0%BD%D0%B8%D0%B9_%D1%88%D0%B8%D1%84%D1%80) і [підстановки](https://uk.wikipedia.org/wiki/%D0%9F%D1%96%D0%B4%D1%81%D1%82%D0%B0%D0%BD%D0%BE%D0%B2%D0%BE%D1%87%D0%BD%D0%B8%D0%B9_%D1%88%D0%B8%D1%84%D1%80). Цикл алгоритму, з яких і складається DES є комбінацією цих технік, коли як об'єкти перемішування виступають біти тексту, ключа і блоків підстановок.

**Початкова перестановка**

На вході подаються 64-бітний блок даних, які переставляються згідно з таблицею:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Початкова**[**перестановка**](https://uk.wikipedia.org/wiki/%D0%9F%D0%B5%D1%80%D0%B5%D1%81%D1%82%D0%B0%D0%BD%D0%BE%D0%B2%D0%BA%D0%B0)**IP** | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 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 |
| 58 | 50 | 42 | 34 | 26 | 18 | 10 | 2 | 60 | 52 | 44 | 36 | 28 | 20 | 12 | 4 | 62 | 54 | 46 | 38 | 30 | 22 | 14 | 6 | 64 | 56 | 48 | 40 | 32 | 24 | 16 | 8 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 | 61 | 62 | 63 | 64 |
| 57 | 49 | 41 | 33 | 25 | 17 | 9 | 1 | 59 | 51 | 43 | 35 | 27 | 19 | 11 | 3 | 61 | 53 | 45 | 37 | 29 | 21 | 13 | 5 | 63 | 55 | 47 | 39 | 31 | 23 | 15 | 7 |

**16 раундів**

Далі 16 разів повторюються наступні операції:

**Функція розширення блоку**

Функція Е розширює 32-бітовий [вектор](https://uk.wikipedia.org/wiki/%D0%92%D0%B5%D0%BA%D1%82%D0%BE%D1%80) {\displaystyle R\_{i-1}} R до 48-бітового вектора E(R) шляхом повторення деяких біт з {\displaystyle R\_{i-1}}R згідно з таблицею:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Функція розширення E** | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| \_ | 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 | \_ |
| 32 | 1 | 2 | 3 | 4 | 5 | 4 | 5 | 6 | 7 | 8 | 9 | 8 | 9 | 10 | 11 | 12 | 13 | 12 | 13 | 14 | 15 | 16 | 17 | 16 | 17 | 18 | 19 | 20 | 21 | 20 | 21 | 22 | 23 | 24 | 25 | 24 | 25 | 26 | 27 | 28 | 29 | 28 | 29 | 30 | 31 | 32 | 1 |

Перший рядок - номери вхідних біт, другий рядок - вихідні біти. Повторення номерів, означає повторення біт.

**Додавання раундового ключа {\displaystyle k\_{i}}**

Блок 48 біт [XOR](https://uk.wikipedia.org/wiki/XOR" \o "XOR)'иться з раундовим ключем {\displaystyle k\_{i}}.

**Таблиці підстановки**

[S-бокси](https://uk.wikipedia.org/wiki/S-%D0%B1%D0%BE%D0%BA%D1%81) мають вигляд:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**S-бокси**](https://uk.wikipedia.org/wiki/S-%D0%B1%D0%BE%D0%BA%D1%81)**{\displaystyle S\_{i}}, i=1…8** | | | | | | | | | | | | | | | | | |
|  | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **0** | **14** | **4** | **13** | **1** | **2** | **15** | **11** | **8** | **3** | **10** | **6** | **12** | **5** | **9** | **0** | **7** |  |
| **1** | **0** | **15** | **7** | **4** | **14** | **2** | **13** | **1** | **10** | **6** | **12** | **11** | **9** | **5** | **3** | **8** | **{\displaystyle S\_{1}}**  S1 |
| **2** | **4** | **1** | **14** | **8** | **13** | **6** | **2** | **11** | **15** | **12** | **9** | **7** | **3** | **10** | **5** | **0** |  |
| **3** | **15** | **12** | **8** | **2** | **4** | **9** | **1** | **7** | **5** | **11** | **3** | **14** | **10** | **0** | **6** | **13** |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **0** | **15** | **1** | **8** | **14** | **6** | **11** | **3** | **4** | **9** | **7** | **2** | **13** | **12** | **0** | **5** | **10** |  |
| **1** | **3** | **13** | **4** | **7** | **15** | **2** | **8** | **14** | **12** | **0** | **1** | **10** | **6** | **9** | **11** | **5** | **{\displaystyle S\_{2}}**  S2 |
| **2** | **0** | **14** | **7** | **11** | **10** | **4** | **13** | **1** | **5** | **8** | **12** | **6** | **9** | **3** | **2** | **15** |  |
| **3** | **13** | **8** | **10** | **1** | **3** | **15** | **4** | **2** | **11** | **6** | **7** | **12** | **0** | **5** | **14** | **9** |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **0** | **10** | **0** | **9** | **14** | **6** | **3** | **15** | **5** | **1** | **13** | **12** | **7** | **11** | **4** | **2** | **8** |  |
| **1** | **13** | **7** | **0** | **9** | **3** | **4** | **6** | **10** | **2** | **8** | **5** | **14** | **12** | **11** | **15** | **1** | **{\displaystyle S\_{3}}**  S3 |
| **2** | **13** | **6** | **4** | **9** | **8** | **15** | **3** | **0** | **11** | **1** | **2** | **12** | **5** | **10** | **14** | **7** |  |
| **3** | **1** | **10** | **13** | **0** | **6** | **9** | **8** | **7** | **4** | **15** | **14** | **3** | **11** | **5** | **2** | **12** |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **0** | **7** | **13** | **14** | **3** | **0** | **6** | **9** | **10** | **1** | **2** | **8** | **5** | **11** | **12** | **4** | **15** |  |
| **1** | **13** | **8** | **11** | **5** | **6** | **15** | **0** | **3** | **4** | **7** | **2** | **12** | **1** | **10** | **14** | **9** | **{\displaystyle S\_{4}}**  S4 |
| **2** | **10** | **6** | **9** | **0** | **12** | **11** | **7** | **13** | **15** | **1** | **3** | **14** | **5** | **2** | **8** | **4** |  |
| **3** | **3** | **15** | **0** | **6** | **10** | **1** | **13** | **8** | **9** | **4** | **5** | **11** | **12** | **7** | **2** | **14** |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **0** | **2** | **12** | **4** | **1** | **7** | **10** | **11** | **6** | **8** | **5** | **3** | **15** | **13** | **0** | **14** | **9** |  |
| **1** | **14** | **11** | **2** | **12** | **4** | **7** | **13** | **1** | **5** | **0** | **15** | **10** | **3** | **9** | **8** | **6** | **{\displaystyle S\_{5}}**  S5 |
| **2** | **4** | **2** | **1** | **11** | **10** | **13** | **7** | **8** | **15** | **9** | **12** | **5** | **6** | **3** | **0** | **14** |  |
| **3** | **11** | **8** | **12** | **7** | **1** | **14** | **2** | **13** | **6** | **15** | **0** | **9** | **10** | **4** | **5** | **3** |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **0** | **12** | **1** | **10** | **15** | **9** | **2** | **6** | **8** | **0** | **13** | **3** | **4** | **14** | **7** | **5** | **11** |  |
| **1** | **10** | **15** | **4** | **2** | **7** | **12** | **9** | **5** | **6** | **1** | **13** | **14** | **0** | **11** | **3** | **8** | **{\displaystyle S\_{6}}**  S6 |
| **2** | **9** | **14** | **15** | **5** | **2** | **8** | **12** | **3** | **7** | **0** | **4** | **10** | **1** | **13** | **11** | **6** |  |
| **3** | **4** | **3** | **2** | **12** | **9** | **5** | **15** | **10** | **11** | **14** | **1** | **7** | **6** | **0** | **8** | **13** |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **0** | **4** | **11** | **2** | **14** | **15** | **0** | **8** | **13** | **3** | **12** | **9** | **7** | **5** | **10** | **6** | **1** |  |
| **1** | **13** | **0** | **11** | **7** | **4** | **9** | **1** | **10** | **14** | **3** | **5** | **12** | **2** | **15** | **8** | **6** | **{\displaystyle S\_{7}}**  S7 |
| **2** | **1** | **4** | **11** | **13** | **12** | **3** | **7** | **14** | **10** | **15** | **6** | **8** | **0** | **5** | **9** | **2** |  |
| **3** | **6** | **11** | **13** | **8** | **1** | **4** | **10** | **7** | **9** | **5** | **0** | **15** | **14** | **2** | **3** | **12** |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **0** | **13** | **2** | **8** | **4** | **6** | **15** | **11** | **1** | **10** | **9** | **3** | **14** | **5** | **0** | **12** | **7** |  |
| **1** | **1** | **15** | **13** | **8** | **10** | **3** | **7** | **4** | **12** | **5** | **6** | **11** | **0** | **14** | **9** | **2** | **{\displaystyle S\_{8}}**  S8 |
| **2** | **7** | **11** | **4** | **1** | **9** | **12** | **14** | **2** | **0** | **6** | **10** | **13** | **15** | **3** | **5** | **8** |  |
| **3** | **2** | **1** | **14** | **7** | **4** | **10** | **8** | **13** | **15** | **12** | **9** | **0** | **3** | **5** | **6** | **11** |  |

"Розширені" біти використовуються для визначення номера 0-1-2-3 таблиці (ліва колонка).

**Перестановка**

Далі виконується перестановка

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Перестановка P** | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 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 |
| 16 | 7 | 20 | 21 | 29 | 12 | 28 | 17 | 1 | 15 | 23 | 26 | 5 | 18 | 31 | 10 | 2 | 8 | 24 | 14 | 32 | 27 | 3 | 9 | 19 | 13 | 30 | 6 | 22 | 11 | 4 | 25 |

[**XOR**](https://uk.wikipedia.org/wiki/XOR)**лівої і правої частин**[[ред.](https://uk.wikipedia.org/w/index.php?title=Data_Encryption_Standard&veaction=edit&section=10) • [ред. код](https://uk.wikipedia.org/w/index.php?title=Data_Encryption_Standard&action=edit&section=10)]

За формулою {\displaystyle R\_{i}=L\_{i-1}\oplus f(R\_{i-1},k\_{i})} отримуємо значення {\displaystyle R\_{i}}Ri

**Кінцева перестановка**

Після 16-ти раундів застосовуюєть перестановка біт:

Вхідний перший біт ставить на місце номер 40, другий біт на місце номер 8 і т.д.

В DES використовується 16 циклів, вихідними даними для кожного з них 64 є біти [відкритого тексту](https://uk.wikipedia.org/wiki/%D0%92%D1%96%D0%B4%D0%BA%D1%80%D0%B8%D1%82%D0%B8%D0%B9_%D1%82%D0%B5%D0%BA%D1%81%D1%82) (на вході першого раунду) чи 64 біти з результату роботи попереднього раунду (у всіх наступних), ключ і блоки підстановок.

Алгоритм використовує лише стандартні елементарні логічні операції (зсув, сума по модулю два([XOR](https://uk.wikipedia.org/wiki/XOR))) на числах довжиною 64 біти, що значно полегшує програмування алгоритму і прискорює його роботу у інтегральних мікросхемах. Циклічний характер алгоритму в сумі з ідеальною здатністю до запрограмовування робить алгоритм швидким і легким до розуміння.

Розширення ключа (генерація раундових ключів {\displaystyle k\_{i}})

Ключі {\displaystyle k\_{i}} отримують з початкового ключа k (64 [біт](https://uk.wikipedia.org/wiki/%D0%91%D1%96%D1%82) = 8 байтів або 8 символів у [ASCII](https://uk.wikipedia.org/wiki/ASCII)) наступним чином. Вісім бітів, що знаходять в позиціях 8, 16, 24, 32, 40, 48, 56, 64 додаються в ключ k таким чином щоб кожен [байт](https://uk.wikipedia.org/wiki/%D0%91%D0%B0%D0%B9%D1%82) містив непарне число одиниць. Це використовується для виявлення помилок при обміні і зберіганні ключів. Потім роблять перестановку для розширеного ключа (крім доданих бітів номер 8, 16, 24, 32, 40, 48, 56, 64). Така перестановка визначена в таблиці 5.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Початкова перестановка біт ключа** | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 57 | 49 | 41 | 33 | 25 | 17 | 9 | 1 | 58 | 50 | 42 | 34 | 26 | 18 | 10 | 2 | 59 | 51 | 43 | 35 | 27 | 19 | 11 | 3 | 60 | 52 | 44 | 36 | {\displaystyle C\_{0}}  C0 |
| 63 | 55 | 47 | 39 | 31 | 23 | 15 | 7 | 62 | 54 | 46 | 38 | 30 | 22 | 14 | 6 | 61 | 53 | 45 | 37 | 29 | 21 | 13 | 5 | 28 | 20 | 12 | 4 | {\displaystyle D\_{0}}  D0 |

Ця перестановка визначається двома блоками {\displaystyle C\_{0}}C0 і D0 по 28 [біт](https://uk.wikipedia.org/wiki/%D0%91%D1%96%D1%82) кожен. Перші 3 біта {\displaystyle C\_{0}}C0 є біти 57, 49, 41 розширеного ключа. А перші три біта {\displaystyle D\_{0}}D D0 є біти 63, 55, 47 розширеного ключа.

Для раундів Ci, Di{\displaystyle C\_{i},D\_{i}},,,,,,,,, i = 1,2,3 ... отримують з {\displaystyle C\_{i-1},D\_{i-1}} Ci-1 , Di-1 одним або двома лівими циклічними зрушеннями згідно з таблицею:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Циклічний зсув** | | | | | | | | | | | | | | | | |
| **i-й раунд** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** |
| Число зсувів | 1 | 1 | 2 | 2 | 2 | 2 | 2 | 2 | 1 | 2 | 2 | 2 | 2 | 2 | 2 | 1 |

Ключ {\displaystyle k\_{i}}ki, i = 1, ... 16 складається з 48 біт, вибраних з бітів вектора {\displaystyle C\_{i}D\_{i}} CiDi (56 [біт](https://uk.wikipedia.org/wiki/%D0%91%D1%96%D1%82)) згідно з приведеною нижче таблицею. Перший і другий біти {\displaystyle k\_{i}}ki є біти 14, 17 вектора {\displaystyle C\_{i}D\_{i}} CiDi

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| i-й біт раундового ключа | 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 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 |
| номери біт 3 векторів {\displaystyle C\_{i}D\_{i}} | 14 | 17 | 11 | 24 | 1 | 5 | 3 | 28 | 15 | 6 | 21 | 10 | 23 | 19 | 12 | 4 | 26 | 8 | 16 | 7 | 27 | 20 | 13 | 2 | 41 | 52 | 31 | 37 | 47 | 55 | 30 | 40 | 51 | 45 | 33 | 48 | 44 | 49 | 39 | 56 | 34 | 53 | 46 | 42 | 50 | 36 | 29 | 32 |

Біти номер 1-48 використовують для раундового [XORу](https://uk.wikipedia.org/wiki/XOR" \o "XOR).

Приклад роботи алгоритму

Для ключа 00 00 00 00 00 00 00 00 (НЕХ)

та тексту 00 00 00 00 00 00 00 00 (НЕХ)

Результати шифрування відкритого тексту ключем

**8C A6 4D E9 C1 B1 23 A7** (НЕХ)

(test vector 8ca64de9c1b123a7)

Детальний приклад

Ключ FEDCBA9876543210 (НЕХ)

Відкритий текст 0123456789ABCDEF (НЕХ)

Результат **ED39D950FA74BCC4** (НЕХ)

Захист та криптоаналіз

Оскільки DES — порівняно старий [криптоалгоритм](https://uk.wikipedia.org/w/index.php?title=%D0%9A%D1%80%D0%B8%D0%BF%D1%82%D0%BE%D0%B0%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC&action=edit&redlink=1" \o "Криптоалгоритм (ще не написана)), існує багато публікацій щодо його [криптоаналізу](https://uk.wikipedia.org/wiki/%D0%9A%D1%80%D0%B8%D0%BF%D1%82%D0%BE%D0%B0%D0%BD%D0%B0%D0%BB%D1%96%D0%B7" \o "Криптоаналіз). Дуже ґрунтовну оцінку безпеки DES дано [Брюсом Шнаєром](https://uk.wikipedia.org/wiki/%D0%91%D1%80%D1%8E%D1%81_%D0%A8%D0%BD%D0%B0%D0%B9%D1%94%D1%80), який у своїй відомій книзі «Прикладна криптографія» розбирає та впорядковує велику кількість публікацій щодо криптоаналізу DES.

Тепер DES вважається нестійким, оскільки:

1. Розмір ключа — 56 бітів — замалий, тому існує реальна загроза пошуку ключа [лобовою атакою](https://uk.wikipedia.org/wiki/%D0%90%D1%82%D0%B0%D0%BA%D0%B0_%D0%BF%D0%BE%D0%B2%D0%BD%D0%BE%D0%B3%D0%BE_%D0%BF%D0%B5%D1%80%D0%B5%D0%B1%D0%BE%D1%80%D1%83) (послідовним перебором).
2. DES нестійкий до [лінійного криптоаналізу](https://uk.wikipedia.org/w/index.php?title=%D0%9B%D1%96%D0%BD%D1%96%D0%B9%D0%BD%D0%B8%D0%B9_%D0%BA%D1%80%D0%B8%D0%BF%D1%82%D0%BE%D0%B0%D0%BD%D0%B0%D0%BB%D1%96%D0%B7&action=edit&redlink=1) (тобто лінійна атака дозволяє знайти ключ DES швидше, ніж послідовний перебір).

В той же час, повний 16-раундовий DES стійкий до [диференційного криптоаналізу](https://uk.wikipedia.org/w/index.php?title=%D0%94%D0%B8%D1%84%D0%B5%D1%80%D0%B5%D0%BD%D1%86%D1%96%D0%B9%D0%BD%D0%B8%D0%B9_%D0%BA%D1%80%D0%B8%D0%BF%D1%82%D0%BE%D0%B0%D0%BD%D0%B0%D0%BB%D1%96%D0%B7&action=edit&redlink=1).

Через високу розповсюдженість DES було запропоновано багато ідей щодо підвищення його безпеки, зокрема, замінити S-блоки DES новими, стійкими до лінійної атаки. Однак, широке практичне застосування жодна з видозмінених версій DES не набула. Винятком є [потрійний DES](https://uk.wikipedia.org/wiki/%D0%9F%D0%BE%D1%82%D1%80%D1%96%D0%B9%D0%BD%D0%B8%D0%B9_DES), однак, це не видозміна алгоритму, а лише особливий режим шифрування звичайним DES.

Код програми

public String Crypt(byte[] data, String key)

{

DESCryptoServiceProvider cryptoProvider = new DESCryptoServiceProvider();

MemoryStream memoryStream = new MemoryStream();

CryptoStream cryptoStream = new CryptoStream(memoryStream,

cryptoProvider.CreateEncryptor(Encoding.ASCII.GetBytes(key), Encoding.ASCII.GetBytes(key)),

CryptoStreamMode.Write);

memoryStream.Write(data,0,data.Length);

cryptoStream.FlushFinalBlock();

var result = Convert.ToBase64String(memoryStream.GetBuffer(), 0, (int)memoryStream.Length);

memoryStream.Close();

cryptoStream.Close();

//writer.Close();

return result;

}

public String Decrypt(byte[] data, String key)

{

DESCryptoServiceProvider cryptoProvider = new DESCryptoServiceProvider();

CryptoStream cryptoStream = new CryptoStream(new MemoryStream(data),

cryptoProvider.CreateDecryptor(Encoding.ASCII.GetBytes(key), Encoding.ASCII.GetBytes(key)), CryptoStreamMode.Read);

StreamReader reader = new StreamReader(cryptoStream);

var result = reader.ReadToEnd();

cryptoStream.Close();

reader.Close();

return result;

}

Висновки:

Було вивчено алгоритм вищевказаного шифру та реалізовано його з відповідністю до його специфікації. Був розроблений дружній інтерфейс користувача для спрощення роботи з програмою.