DES Documentation

Made with Love, by Mousa Emarah

1. Key Generation Process:

1.1 Initial Permutation (PC1):

- The 64-bit key undergoes permutation using the PC1 table
- Results in a 56-bit key

1.2 Key Division:

- The 56-bit key is divided into two equal halves:
  - Left half (C_0): 28 bits
  - Right half (D_0): 28 bits

1.3 Subkey Creation:

- Sixteen subkeys are generated through 16 rounds
- Each round applies circular left shifts to both halves
- Shift amounts follow the predefined shift schedule
- Each shifted pair forms a 56-bit intermediate key

1.4 Final Subkey Permutation (PC2):

- Each 56-bit intermediate key is permuted using PC2 table
- Produces 16 final 48-bit subkeys (K_1 to K_{16})
- These subkeys are used in the Feistel network

2. Message Encryption Process:

2.1 Input Conversion:

- Plaintext message is converted to 64-bit binary representation
- Padding applied if message is not 64 bits

2.2 Initial Permutation (IP):

- 64-bit block undergoes permutation using IP table
- Reorders the bits according to IP table specification

2.3 Block Division:

- Permuted block is split into:
  - Left half (L_0): 32 bits
  - Right half (R_0): 32 bits

2.4 Feistel Rounds (16 iterations):

For each round n (1 to 16):

2.4.1 $L_n = R_{n-1}$

2.4.2 $R_n = L_{n-1} \oplus f(R_{n-1}, K_n)$

Where:

- $\oplus$ = XOR operation
- f = Feistel function
- K_n = Subkey for round n

2.5 Expansion (E):

- Right half (R_{n-1}) expanded from 32 to 48 bits
- Uses expansion table E
- Allows combination with 48-bit subkey

2.6 S-box Substitution:

- 48-bit result divided into eight 6-bit groups
- Each group processed through corresponding S-box (S1-S8)
- Each S-box outputs 4 bits
- Total output: 32 bits (8 × 4 bits)

2.7 P Permutation:

- 32-bit S-box output undergoes permutation
- Uses P table to rearrange bits
- Produces final output of Feistel function

2.8 XOR Operation:

- P-permuted result is XORed with L_{n-1}
- Produces new right half R_n
- Left half L_n becomes previous R_{n-1}

3. Final Steps:

3.1 After 16 rounds:

- Left and right halves are concatenated
- Note: No swap after final round

3.2 Final Permutation (FP):

- Combined block undergoes final permutation
- Uses inverse of initial permutation table
- Produces final 64-bit ciphertext

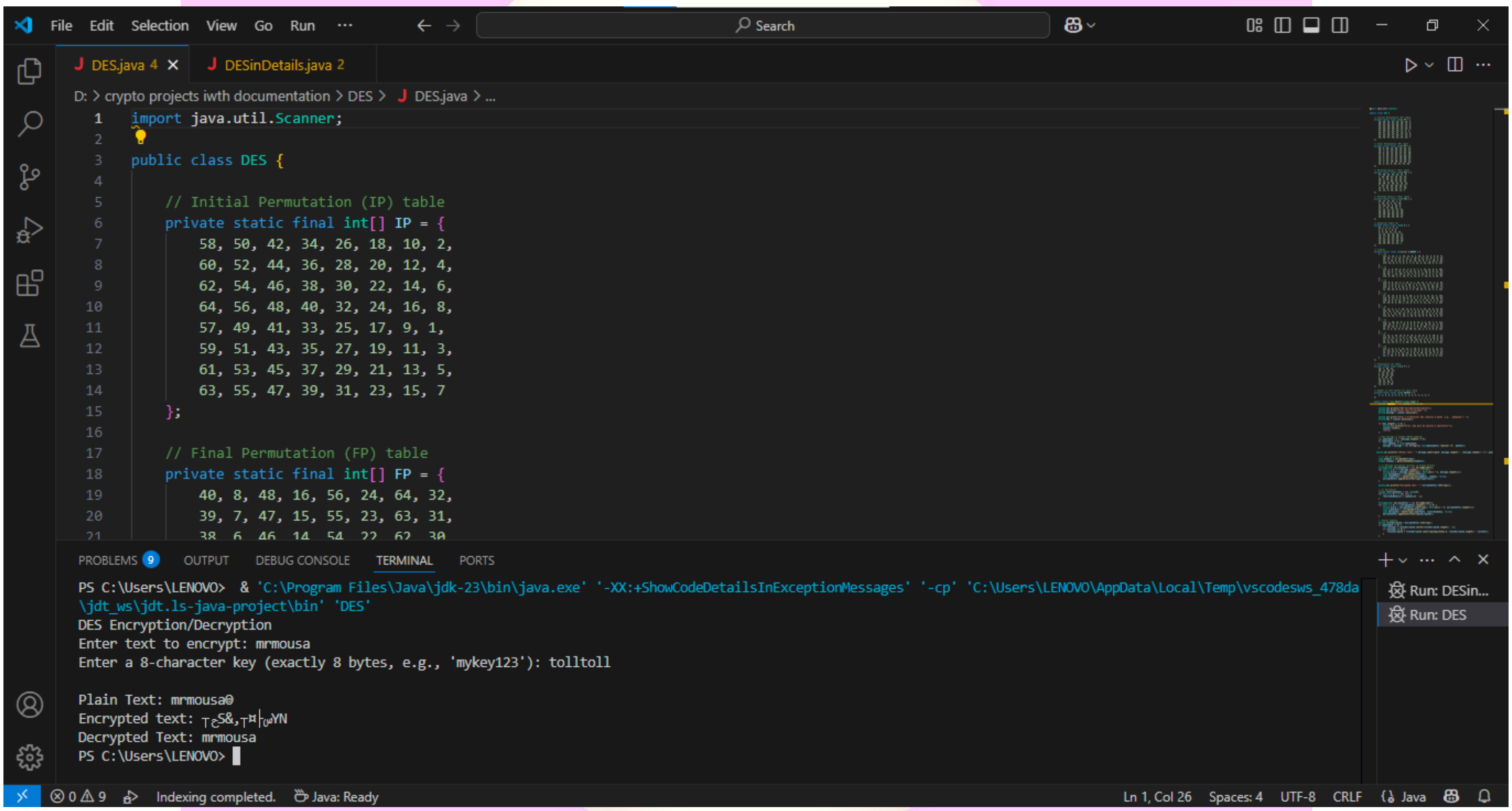
https://linkedin.com/in/mousa123

4. Decryption Process:

4.1 Same algorithm as encryption

4.2 Subkeys used in reverse order (K_{16} to K_1)

4.3 Final output is original plaintext



Always remember:

1)Key generation:

1- permutation using pc1 table.
2- divide each key into 2 each of 28 bits.
3- now we have 16 subkeys "56 bits"
4- therefore apply pc2 on contacanated subkeys to have 48 bits to work in fesitel network.
(These enter the function with R0 then XORed with L0 to get R1)

2)Message encryption:

1-change the message inputed by the user into binary
2- apply ip permutation
3-divide into 2 halves
4- Ln = Rn-1 , Rn = Ln-1 +f(Rn-1 , Kn) , where n is the number of iteration
5- use expansion permutation no have 8 groups of 6 bits to have 48 bits with the 48 bits of the keys
6-Apply s boxes (now we have 8 groups of 4 bits)
7-Apply P permuation which gives us the ifnal value of f
8- xor with the Ln-1

https://linkedin.com/in/mousa123

Reference:
https://page.math.tu-berlin.de/~kant/teaching/hess/krypto-ws2006/des.htm