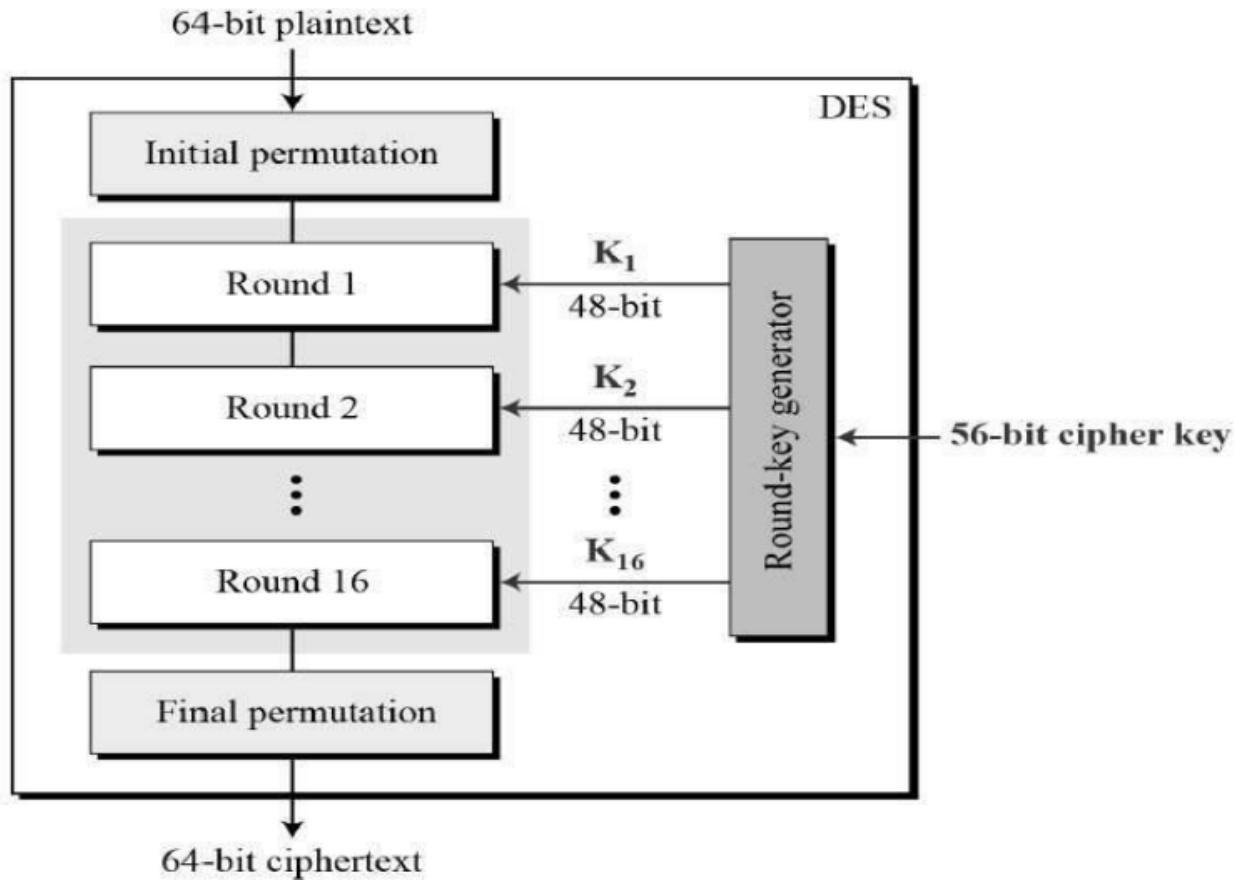
ASSIGNMENT 6

Write a program to implement Data Encryption Standard for encryption and decryption.



1. Process the key.
    i. Get a 64-bit key from the user.
    ii. Calculate the key schedule.
        1. Perform the following permutation on the 64-bit key. The parity bits are discarded, reducing the key to 56 bits. Bit 1 of the permuted block is bit 57 of the original key, bit 2 is bit 49, and so on with bit 56 being bit 4 of the original key.
        2. Split the permuted key into two halves. The first 28 bits are called C[0] and the last 28 bits are called D[0].
        3. Calculate the 16 subkeys. Start with i = 1.
            1. Perform one or two circular left shifts on both C[i-1] and D[i-1] to get C[i] and D[i], respectively. The number of shifts per iteration are given in the table below.
            Iteration # 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
            Left Shifts 1 1 2 2 2 2 2 2 1 2 2 2 2 2 2 1
            2. Permute the concatenation C[i]D[i] as indicated below. This will yield K[i], which is 48 bits long.
            3. Loop back to 1.ii.c.1 until K[16] has been calculated.

2. Process a 64-bit data block.
i. Get a 64-bit data block. If the block is shorter than 64 bits, it should be padded as appropriate for the application.
ii. Perform the initial permutation on the data block.

iii. Split the block into two halves. The first 32 bits are called L[0], and the last 32 bits are called R[0].
iv. Apply the 16 subkeys to the data block. Start with i = 1.

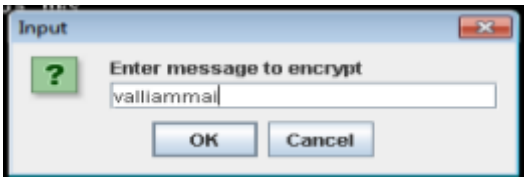
a. Expand the 32-bit R[i-1] into 48 bits according to the bit-selection function Expansion (E)
b. Exclusive-or E(R[i-1]) with K[i].
c. Break E(R[i-1]) xor K[i] into eight 6-bit blocks. Bits 1-6 are B[1], bits 7-12 are B[2], and so on with bits 43-48 being B[8].
d. Substitute the values found in the S-boxes for all B[j]. Start with j = 1. All values in the S-boxes should be considered 4 bits wide.

i. Take the 1st and 6th bits of B[j] together as a 2-bit value (call it m) indicating the row in S[j] to look in for the substitution.
ii. Take the 2nd through 5th bits of B[j] together as a 4-bit value(call it n) indicating the column in S[j] to find the substitution.
iii. Replace B[j] with S[j][m][n].
iv. Loop back to 2.iv.d.i until all 8 blocks have been replaced.

e. Permute the concatenation of B[1] through B[8]
f. Exclusive-or the resulting value with L[i-1]. Thus, all together, your R[i] = L[i-1] xor P(S[1](B[1])...S[8](B[8])), where B[j] is a 6-bit block of E(R[i-1]) xor K[i]. (The function for R[i] is written as, R[i] = L[i-1] xor f(R[i-1], K[i]).)
g. L[i] = R[i-1].
h. Loop back to 2.iv.a until K[16] has been applied.
v. Perform the final permutation on the block R[16]L[16].

3. Decryption : Use the keys K[i] in reverse order. That is, instead of applying K[1] for the first iteration, apply K[16], and then K[15] for the second, on down to K[1]

Input:



Output: