**Marathwada Mitra Mandal’s**

**COLLEGE OF ENGINEERING, PUNE**

**DEPARTMENT OF COMPUTER ENGINEERING**

**BE Comp-II 2020-21**

**SUBJECT: ICS SEMESTER-II**

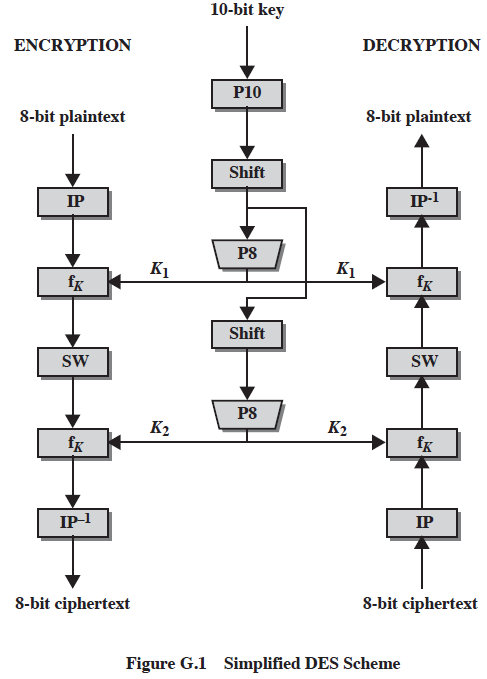
**Lab Assignment1**

**Implementation of S-DES algorithm**

**Group Project by Dibyo,Durvesh**

**Theory:**

Figure G.1 illustrates the overall structure of the Simplified DES, which we will refer to as S-DES.The S-DES encryption algorithm takes an 8-bit block of plaintext (example: 10111101) and a 10-bit key as input and produces an 8-bit block of cipher text as output. The S-DES decryption algorithm takes an 8-bit block of cipher text and the same 10-bit key used to produce that cipher text as input and produces the original 8-bit block of plaintext. The encryption algorithm involves five functions: an initial permutation (IP); a complex function labelled fK, which involves both permutation and substitution operations and depends on a key input; a simple permutation function that switches (SW) the two halves of the data; the function fK again; and finally a permutation function that is the inverse of the initial permutation (IP–1). The function fK takes as input not only the data passing through the encryption algorithm, but also an 8-bit key. The algorithm could have been designed to work with a 16-bit key, consisting of two 8-bit subkeys, one used for each occurrence of fK. Alternatively, a single 8-bit key could have been used, with the same key used twice in the algorithm. A compromise is to use a 10-bit key from which two 8-bit subkeys are generated, as depicted in Figure G.1. In this case, the key is first subjected to a permutation (P10). Then a shift operation is performed. The output of the shift operation then passes through a permutation function that reduces an 8-bit output (P8) for the first subkey (K1). The output of the shift operation also feeds into another shift and another instance of P8 to produce the second subkey (K2).



**DES algorithm steps:**

**DES pseudocode for**

**implementation:**

**public** **byte** encrypt( **int** m)

{

System.***out***.println("\nEncryption Process Starts........\n\n");

m = *permute*( m, ***IP***, ***IPmax***);

System.***out***.print("\nAfter Initial Permutation(IP) : ");

*printData*( m, 8);

m = *fK*( m, K1);

System.***out***.print("\nbefore Swap : ");

*printData*( m, 8);

m = *SW*( m);

System.***out***.print("\nAfter Swap : ");

*printData*( m, 8);

m = *fK*( m, K2);

System.***out***.print("\nbefore IP inverse : ");

*printData*( m, 8);

m = *permute*( m, ***IPI***, ***IPImax***);

**return** (**byte**) m;

}

**public** **byte** decrypt( **int** m)

{

System.***out***.println("\nDecryption Process Starts........\n\n");

*printData*( m, 8);

m = *permute*( m, ***IP***, ***IPmax***);

System.***out***.print("\nAfter Permutation : ");

*printData*( m, 8);

m = *fK*( m, K2);

System.***out***.print("\nbefore Swap : ");

*printData*( m, 8);

m = *SW*( m);

System.***out***.print("\nAfter Swap : ");

*printData*( m, 8);

m = *fK*( m, K1);

System.***out***.print("\nBefore Extraction Permutation : ");

*printData*( m, 4);

m = *permute*( m, ***IPI***, ***IPImax***);

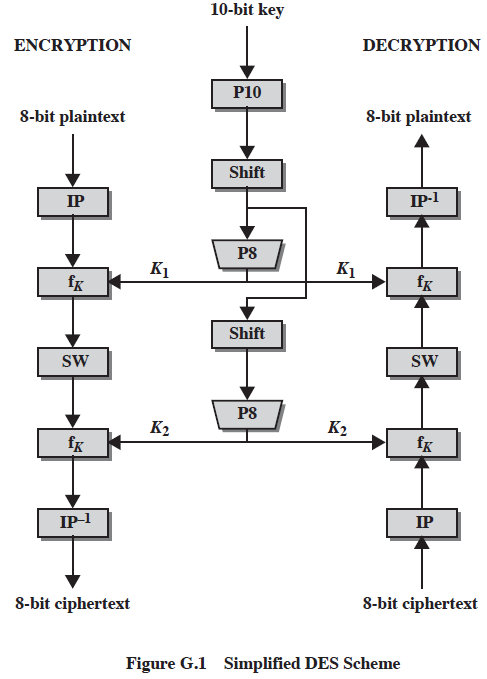
System.***out***.print("\nAfter Extraction Permutation : ");

*printData*( m, 8);

**return** (**byte**) m;

}

**Implementation steps:**



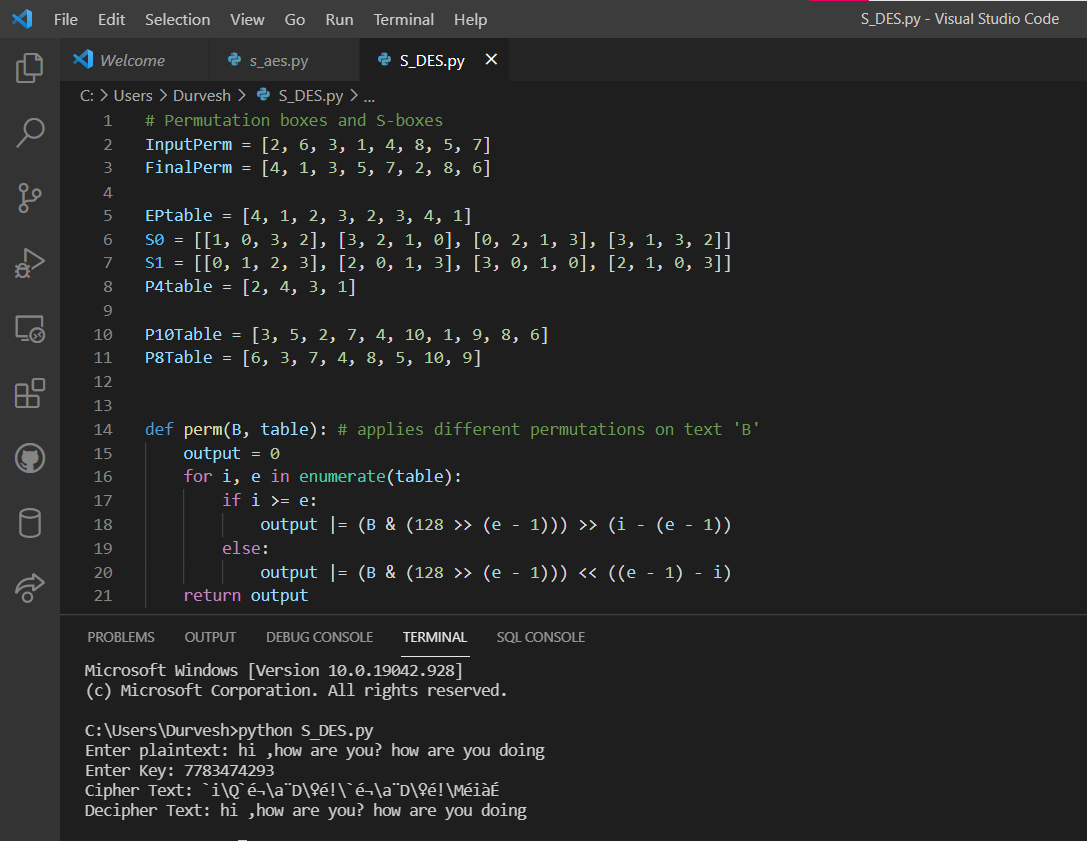
**Result with screenshot:**

**Enter plaintext: hi,how are you?**

**Enter Key: 7373788924**

**Cipher Text: `iQ`é¬\a¨D\♀é!**

**Decipher Text: hi,how are you?**

****

**Outcome :**

**Thus the implementation of S-DES algorithm was successfully conducted.**