Peter Klutchko

CISC 3580

Professor Ammari

16 November 2018

Project 2 Report

This project is a DES program. The program generates an array to represent the 64 bits in the DES plaintext and fills the array with random letters. A key of the same size is also created at the same time and filled with random numbers. However, the program only generates the key and plaintext through user input. User input is obtained through an interface where the user may input the numbers 0-5 to either exit the program, generate a new plaintext and key, encrypt the plaintext, output the plaintext only, output the key only, or output the ciphertext only.

The program itself is divided into four functions: the main function, changeLetter, respOne, and respTwo. The main function holds the code for the interface and calls to the respOne and respTwo functions. respOne is called should the user input a response of ‘1’ into the program. In the function the plaintext and key are generated. Inside a for-loop the plaintext array, “DES”, is assigned a random number from 0 to 25. A switch statement will assign a letter to “DES” depending on the number generated. The number 0 will generate “a” and 1 will generate “b” and so on and so forth. respTwo is called should the user input a response of ‘2’ into the program. This function is what encrypts the DES array. However, the array itself is not changed but instead copied to another array which will be passed to hold the ciphertext. In respTwo the constant integer “USABLE\_KEY” of size 48 represents the numbers of the original key that will be used to encrypt both halves of the DES array. The DES array is then split up into two different arrays, “leftSide” and “rightSide”, for each round of encryption. A for-loop will run 16 times and change which side it encrypts every other run. The encryption process involves invoking a function called changeLetter, which does exactly what the name suggests; the function will change the letter of a char passed to it. Since the for-loop runs 16 times, this function will be called 48 x 16 times (once for every letter in either array multiplied by how many times the for-loop will run). By the end of the for-loop the array should be encrypted and the program will output the result.