Assignment

Programming project: Simplified DES, part 2

30 points

Continue your implementation of SDES. Call this program **`SDESEncrypt.java`** or **`sdesencrypt.py`** and write the program so that it contains a method the method **`encrypt(plaintext, key)`**, which returns an integer ciphertext. Write the program so that it can be run from the command line taking two parameters:

1. the key integer
2. the plaintext integer

Note that these parameters are given in the opposite order from the previous assignment! When run, the program should print the ciphertext as a single integer. In other words, the program must be stand-alone.

Generating the key schedule Indexing bits correctly is important for deriving the sub-keys. The text does not make it clear how that's done for SDES. Please use this method as this is what I used in generating the test results. The bits in a string are numbered left-to-right start from 1. From the 9-bit key, the sub-key for round *i* is found by taking the eight bits starting at index *i* in the key, wrapping around if necessary. To make that clear, here are the index sequences for each sub-key:

Round	Sequence	Subkeys for key 011111111
1	12345678	01111111
2	23456789	11111111
3	34567891	11111110
4	45678912	11111101

My results are in this table. As already noted, it's easy to get this wrong! If your results depart from these, let me know. As several people pointed out last, I had an error in my implemenation.

Plaintext	Key	Ciphertext
0	0	1323 (010100101011)
0	85	1097 (010001001001)
0	170	1973 (011110110101)
0	255	599 (001001010111)
0	341	3631 (111000101111)
0	511	1726 (011010111110)
1365	0	2249 (100011001001)

1365	85	2699 (101010001011)
1365	170	2799 (101011101111)
1365	255	1567 (011000011111)
1365	341	1867 (011101001011)
1365	511	769 (001100000001)
2730	0	3326 (110011111110)
2730	85	2086 (100000100110)
2730	170	2228 (100010110100)
2730	255	3266 (110011000010)
2730	341	1296 (010100010000)
2730	511	1846 (011100110110)
4095	0	2369 (100101000001)
4095	85	317 (000100111101)
4095	170	464 (000111010000)
4095	255	1793 (011100000001)
4095	341	2122 (100001001010)
4095	511	2772 (101011010100)

Submission instructions

Please make sure your name appears in comments at the top of the source file and that the program file is given the name I specifed above. Not doing either of these will cost points. Submit the .java or .py file (not zipped!) to the D2L submission folder provided for it.

Document last updated on February 14th, 2021.