The University of Newcastle
School of Information and Physical Sciences

COMP3260 Data Security

Assignment 2

This assignment is to be done in pairs

*Due on **Sunday, 15 May 2022, 11:59pm**, electronically via the "Assignment 2" submission link in Canvas.*

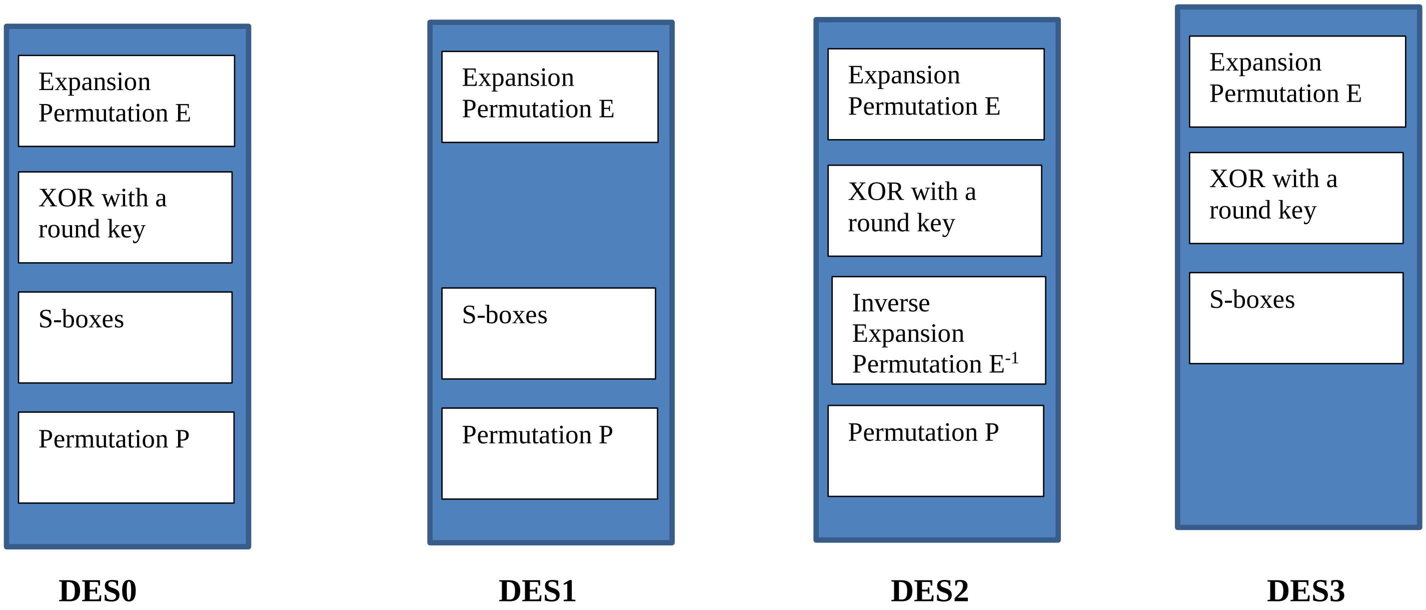
Total 100 marks

Note:
Before you start working on the assignment, please read the information on academic integrity, which can be found at http://www.newcastle.edu.au/service/academic-integrity/. All available strategies will be used for detecting possible plagiarism and all suspicious cases will be referred to the SACO (Student Academic Conduct Officer).

In this assignment, you will implement DES encryption and decryption of a single plaintext block. Your program will take as input a 64-bit plaintext block and a 64-bit key block (note that only 56 bits of those will be selected by PC-1) and produce as output a 64-bit ciphertext block. You will use your implementation to explore the Avalanche effect of the original DES denoted as DES0, as well as DES1, DES2, and DES3, where in each version an operation is missing in each round as follows:

0. DES0 - the original version of DES
1. DES1 – XOR with a round key is missing from F function in all rounds
2. DES2 – S-boxes are missing from F function in all rounds; instead, inverse E^{-1} of the Expansion Permutation E is used for contraction from 48 bits down to 32 bits
3. DES3 – Permutation P is missing from F function in all rounds

For additional clarity, the encryption algorithm for the four versions of DES is given in the picture bellow.



In addition to the original plaintext block P and the key K, your program should use another plaintext blocks P', such that P' differs from P only in (any of) **one bit** and another K' differs from K only in (any of) **one bit**, and use them to explore the Avalanche effect in DES as follows.

The program will encrypt plaintext P under key K. Then it will encrypt plaintext P' under key K and it will find the number of different bits after each of the 16 rounds between P under K, and P' under K.

Similarly, your program will encrypt plaintext P under key K' and it will find the number of different bits after each of the 16 rounds between P under K, and P under K'.

Your program MUST be well commented, include a header stating the authors and purpose of the program, and be easy to understand. You MUST NOT use any available DES code or a portion of it.

Encryption

INPUT FILE

The following is an example of an input file, where

- the first row is the plaintext P
- the second row is the plaintext P'
- the third row is key K
- the last row is key K'

```
000…0
010…0
111…0
110…0
```

OUTPUT FILE

The following is a format of an output file (note that the numbers provided are sample values and not necessarily what you will obtain for different inputs):

```
Avalanche Demonstration
Plaintext P:  000…0
Plaintext P’: 010…0
Key K:  111…0
Key K’: 110…0
Total running time: XXX (second)

P and P’ under K
Ciphertext C:  010…0
Ciphertext C’: 101…1
Round          DES0     DES1     DES2     DES3
   0             1        1        1        1
   1             5       etc
```

2	20
3	30
4	31
5	34
6	32
7	29
8	36
9	41
10	38
11	29
12	33
13	39
14	36
15	40
16	37

P under K and K’

Ciphertext C: 110…1

Ciphertext C’: 001…0

Round	DES0	DES1	DES2	DES3
0	0	etc		
1	2			
2	18			
3	27			
4	33			
5	41			
6	30			
7	34			
8	37			
9	29			
10	33			
11	40			
12	37			
13	43			
14	38			
15	29			
16	35			

In the above, ‘Round 0’ refers to the plain text before the beginning of the encryption. The column DESi contains the number of bits that differ between the original plaintext P (resp. the original key K), and the intermediate result in each round of the encryption performed by DESi defined above.

Decryption

For decryption, the INPUT FILE should contain the ciphertext and the key, and the OUTPUT FILE should contain the ciphertext, the key and the plaintext.

The following is an example of an input file, where

- the first row is the ciphertext C
- the second row is the original key K

```
000…0
111…0
```

The following is a format of an output file (note that the numbers provided are sample values and not necessarily what you will obtain for different inputs):

```
DECRYPTION
Ciphertext C: 000…0
Key K: 111…0
Plaintext P: 010…0
```

Program Requirements

This assignment may be completed in Python, Java or C++. If you would like to use another programming language, please first obtain a permission from your marker Cody Lewis (cody.lewis@newcastle.edu.au).

Submission

All assignments must be submitted via Canvas. If you submit more than once, then only the latest will be graded. Your submission should be one ZIP file containing:

- A PDF file that contains outputs of your program
- All source code files
- A text README file that contains instructions to execute your code

The mark for an assessment item submitted after the designated time on the due date, without an approved extension of time, will be reduced by 10% of the possible maximum mark for that assessment item for each day or part day that the assessment item is late. Note: this applies equally to week and weekend days.

Assessment criteria

1	DES encryption and decryption – working and correct	60
2	Avalanche analysis, correct	30
3	Comments throughout the program	10
	TOTAL	100

If your DES encryption and decryption are not working correctly you can score at most 40 marks in total.