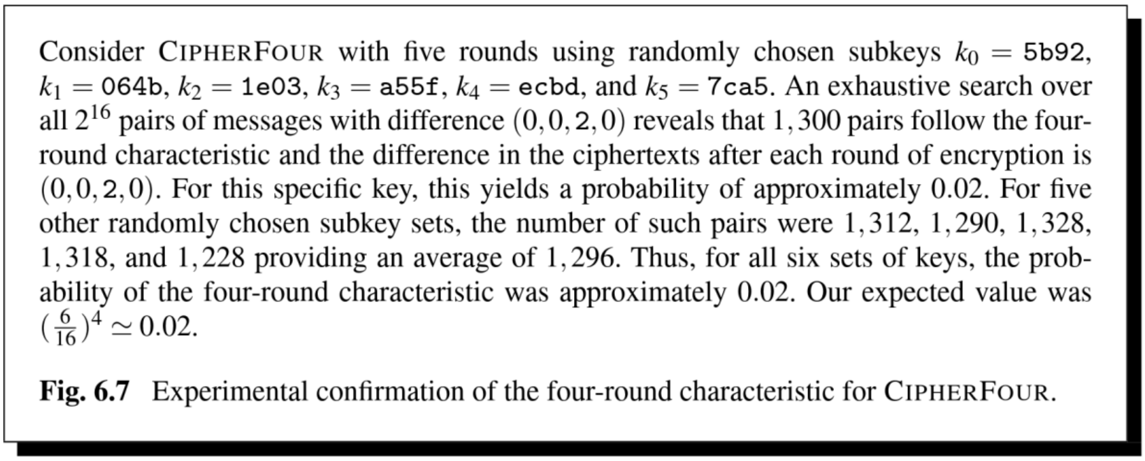
Question Number 5

Block Cipher Cryptanalysis

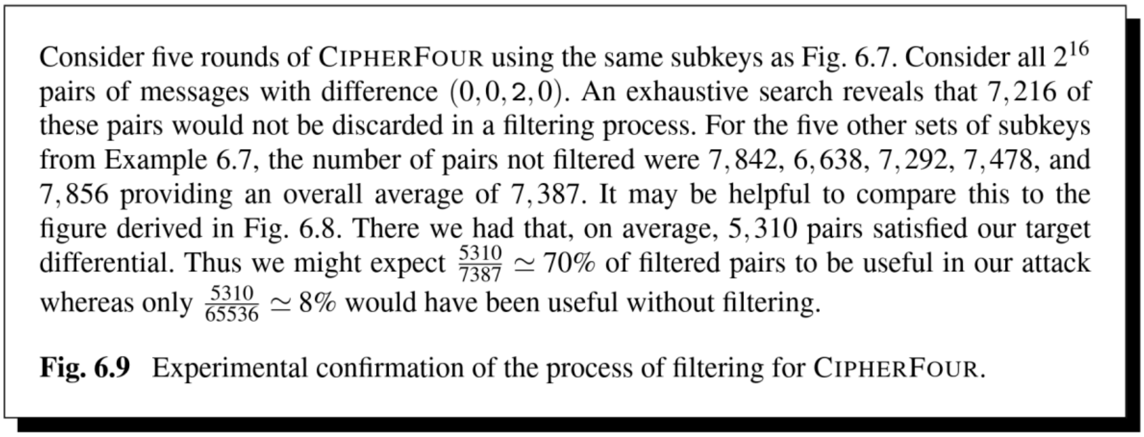
Recall the experiments from Chapter 6 of the "The Block Cipher Companion Book" discussed in class.

- Implement Sypher0041
- Use random keys using openssl rand
  - (Six 16-bits keys → k0 , · · · , k5)
- Verify the following experiments from the book using your implementation of Sypher004.
- Use 6 sets of randomly generated keys.
- Submit the well commented code so that your results can be verified.
- Also document (preferably in a table) the randomly generated keys used here.
- Write paragraphs for each one to describe your observations referring to the figures below.



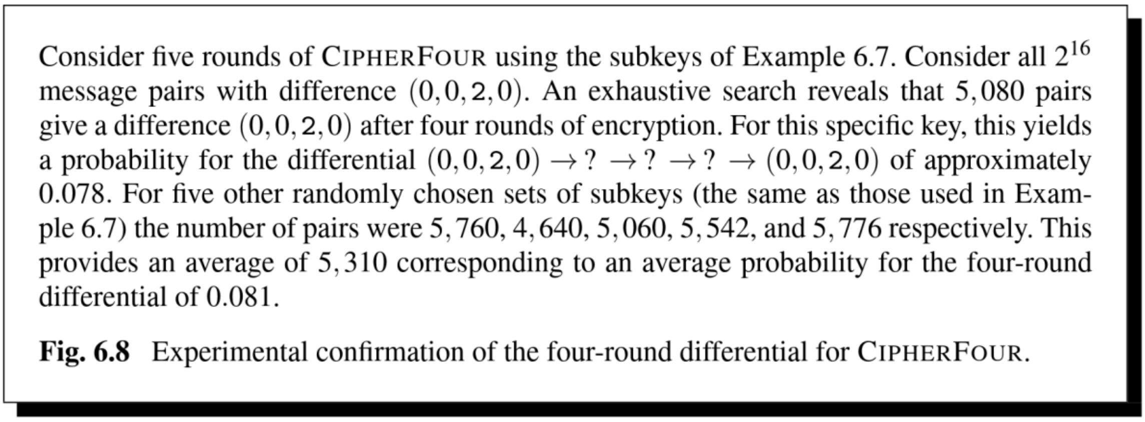
Consider CipherFour with five rounds using randomly chosen subkeys $k_0 = \texttt{5b92}$, $k_1 = \texttt{064b}$, $k_2 = \texttt{1e03}$, $k_3 = \texttt{a55f}$, $k_4 = \texttt{ecbd}$, and $k_5 = \texttt{7ca5}$. An exhaustive search over all 2^{16} pairs of messages with difference $(0,0,2,0)$ reveals that 1,300 pairs follow the four-round characteristic and the difference in the ciphertexts after each round of encryption is $(0,0,2,0)$. For this specific key, this yields a probability of approximately 0.02. For five other randomly chosen subkey sets, the number of such pairs were 1,312, 1,290, 1,328, 1,318, and 1,228 providing an average of 1,296. Thus, for all six sets of keys, the probability of the four-round characteristic was approximately 0.02. Our expected value was $(\frac{6}{16})^4 \simeq 0.02$.

Fig. 6.7 Experimental confirmation of the four-round characteristic for CipherFour.



Consider five rounds of CipherFour using the same subkeys as Fig. 6.7. Consider all 2^{16} pairs of messages with difference $(0,0,2,0)$. An exhaustive search reveals that 7,216 of these pairs would not be discarded in a filtering process. For the five other sets of subkeys from Example 6.7, the number of pairs not filtered were 7,842, 6,638, 7,292, 7,478, and 7,856 providing an overall average of 7,387. It may be helpful to compare this to the figure derived in Fig. 6.8. There we had that, on average, 5,310 pairs satisfied our target differential. Thus we might expect $\frac{5310}{7387} \simeq 70\%$ of filtered pairs to be useful in our attack whereas only $\frac{5310}{65536} \simeq 8\%$ would have been useful without filtering.

Fig. 6.9 Experimental confirmation of the process of filtering for CipherFour.



Consider five rounds of CIPHERFOUR using the subkeys of Example 6.7. Consider all 2^{16} message pairs with difference $(0,0,2,0)$. An exhaustive search reveals that 5,080 pairs give a difference $(0,0,2,0)$ after four rounds of encryption. For this specific key, this yields a probability for the differential $(0,0,2,0) \rightarrow ? \rightarrow ? \rightarrow ? \rightarrow (0,0,2,0)$ of approximately 0.078. For five other randomly chosen sets of subkeys (the same as those used in Example 6.7) the number of pairs were 5,760, 4,640, 5,060, 5,542, and 5,776 respectively. This provides an average of 5,310 corresponding to an average probability for the four-round differential of 0.081.

Fig. 6.8 Experimental confirmation of the four-round differential for CIPHERFOUR.

Solution.

The results of cryptanalysis experiment on a 4-round block cipher, Sypher004, as described in Chapter 6 of *The Block Cipher Companion*. The experiments verify the theoretical probabilities of various ciphertext characteristics after multiple rounds of encryption, using randomly generated subkeys. Three key experiments were conducted, and the results are analyzed and compared with theoretical expectations.

Code Overview

- **sypher004.py**: this file contains the implementation of Sypher004 using `S_Box` and `P_Box`. Implementation done in `Sypher004` function which is returning all round output as we need it in experiments by taking message and key_list as argument. So to get ciphertext corresponding to message given in input to function, we need to take the last of item of the list return by the function.
- **utils.py**: This files mainly contain the utility function as follows: -
  a) `generate_message_pairs`: This will create 2^{16} a message pair with difference $(0, 0, 2, 0)$.
  b) `generate_random_keys`: This function will create random 6 keys of width 16 bits with `openssl`
  c) `printResult`: This function will print data in the form of table
- **perfrom_all_experiments.py**: This is main file where all individual experiment called through respective file and will run simultaneously. Observe the output of this file for all experiments.

Experiment 1: Verification of 4-Round Characteristic (Figure 6.7)

The first experiment aims to confirm that pairs of messages with an initial difference $(0, 0, 2, 0)$follow the same difference after each round of encryption in Sypher004. Specifically, we want to verify that the difference between the ciphertexts remains $(0, 0, 2, 0)$ after all four rounds.

- **Code Overview** The file `experiment_6_7.py` contains all code for this experiment. First, it will get all round outputs as an array and then check the output difference in each round is (0,0,2,0) along with ciphertext. And if the condition is getting satisfied, then increasing count variable. Like this, iterating all over 2^{16} message pairs returning a result as count, probabilities and corresponding key list.

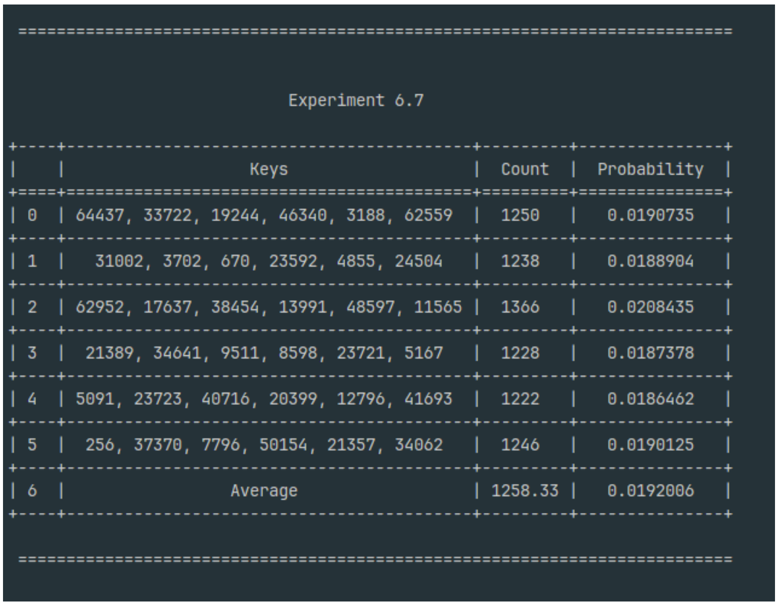
  Theoretically number of message pair follow this conditions should $(\frac{6}{16})^4 \times (2^{16})$ which is nearly 1300

- **output**

Keys	Count	Probability
64437, 33722, 19244, 46340, 3188, 62559	1250	0.0190735
31002, 3702, 670, 23592, 4855, 24504	1238	0.0188904
62952, 17637, 38544, 13991, 48597, 11565	1366	0.0200435
21389, 34641, 9511, 8598, 23721, 5167	1228	0.0187378
5091, 23723, 40716, 20399, 12796, 41693	1222	0.0186462
256, 23770, 7796, 50154, 21357, 34062	1246	0.0190125
Average	1258.33	0.0192006

- **Observation** The results align with the theoretical probability of approximately 0.02. For each set of random subways, around 1300 message pairs followed the expected characteristic, confirming the expected differential cryptanalysis behavior for the 4-round block cipher.

- **Output Image**



```
 ==========================================================================


                              Experiment 6.7

+----+------------------------------------------+---------+---------------+
|    |                   Keys                   |  Count  |  Probability  |
+====+==========================================+=========+===============+
| 0  | 64437, 33722, 19244, 46340, 3188, 62559  |  1250   |   0.0190735   |
+----+------------------------------------------+---------+---------------+
| 1  |   31002, 3702, 670, 23592, 4855, 24504   |  1238   |   0.0188904   |
+----+------------------------------------------+---------+---------------+
| 2  | 62952, 17637, 38454, 13991, 48597, 11565 |  1366   |   0.0208435   |
+----+------------------------------------------+---------+---------------+
| 3  |  21389, 34641, 9511, 8598, 23721, 5167   |  1228   |   0.0187378   |
+----+------------------------------------------+---------+---------------+
| 4  | 5091, 23723, 40716, 20399, 12796, 41693  |  1222   |   0.0186462   |
+----+------------------------------------------+---------+---------------+
| 5  |  256, 37370, 7796, 50154, 21357, 34062   |  1246   |   0.0190125   |
+----+------------------------------------------+---------+---------------+
| 6  |                 Average                  | 1258.33 |   0.0192006   |
+----+------------------------------------------+---------+---------------+


 ==========================================================================
```

Experiment 2: Verification of 4-Round Differential (Figure 6.8)

The second experiment verifies the probability that pairs of messages with an input difference $(0, 0, 2, 0)$will have the same difference after four rounds of encryption. The Difference is here, we are not caring about the difference in intermediate rounds. So this is Possible in 4 ways as described in a book.

Theoretically, the number of message pairs following these conditions should be $4 \times (\frac{6}{16})^4 \times (2^{16})$ which is nearly equal to 5200

- **Code Overview** The file `experiment_6_8.py` contains all code for this experiment. First, it will get all round outputs as an array and then check the output difference only in the last round (0,0,2,0). And if the condition is getting satisfied, then increasing count variable. Like this, iterating all over 2^{16} message pairs returning a result as count, probabilities and corresponding key list.

  The function `isOutDiff_2` checks whether the output difference between two rounds is $(0, 0, 2, 0)$ (hexadecimal difference of 32). The experiment runs over 2^{16} message pairs for each of six sets of keys.

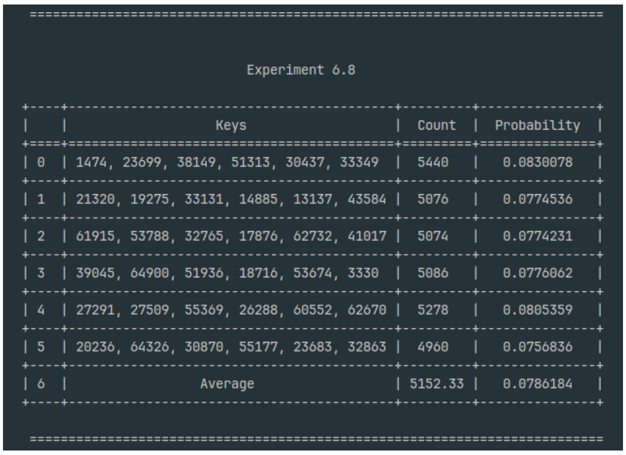
- **Output**

  The results for six sets of keys are shown in the Table. The probability is calculated as the ratio of the valid pairs to the total number of pairs.

Keys	**Count**	**Probability**
1474, 23699, 38149, 51313, 30437, 33349	5440	0.0830078
21320, 19275, 33131, 14885, 13137, 43584	5076	0.0774536
61915, 53788, 32765, 17876, 62732, 41017	5074	0.0774231
39045, 64900, 51936, 18716, 53674, 3330	5086	0.0776062
27291, 27509, 55369, 26288, 60552, 62670	5278	0.0805359
20236, 64326, 30870, 55177, 23683, 32863	4960	0.0756834
Average	5152.33	0.0786184

- **Observation** The probability of the differential being followed after four rounds is approximately 0.081, matching the theoretical expectations. This confirms that for the 4-round Sypher004, the differential propagates with an approximate probability of 0.08.

- **Output Image**



```
 =========================================================================


                              Experiment 6.8

+----+------------------------------------------+---------+---------------+
|    |                   Keys                   |  Count  |  Probability  |
+====+==========================================+=========+===============+
| 0  | 1474, 23699, 38149, 51313, 30437, 33349  | 5440    |   0.0830078   |
+----+------------------------------------------+---------+---------------+
| 1  | 21320, 19275, 33131, 14885, 13137, 43584 | 5076    |   0.0774536   |
+----+------------------------------------------+---------+---------------+
| 2  | 61915, 53788, 32765, 17876, 62732, 41017 | 5074    |   0.0774231   |
+----+------------------------------------------+---------+---------------+
| 3  | 39045, 64900, 51936, 18716, 53674, 3330  | 5086    |   0.0776062   |
+----+------------------------------------------+---------+---------------+
| 4  | 27291, 27509, 55369, 26288, 60552, 62670 | 5278    |   0.0805359   |
+----+------------------------------------------+---------+---------------+
| 5  | 20236, 64326, 30870, 55177, 23683, 32863 | 4960    |   0.0756836   |
+----+------------------------------------------+---------+---------------+
| 6  |                 Average                  | 5152.33 |   0.0786184   |
+----+------------------------------------------+---------+---------------+


 =========================================================================
```

Experiment 3: Filtering Process Verification (Figure 6.9)

This experiment examines how many pairs of messages survive the filtering process during a cryptanalysis attack based on their output differences.

Theoretically number of message pairs follow these conditions should be around 7200

- **Code Overview** The file `experiment_6_9.py` contain all code for this experiment. First, it will get all ciphertext outputs and then check output is in format $(0, 0, *, 0)$ where $*\epsilon(1, 2, 9, a)$. And if condition is getting satisfied then increasing count variable. Like this, iterating all over 2^{16} message pairs returning a result as count, probabilities and corresponding key list.

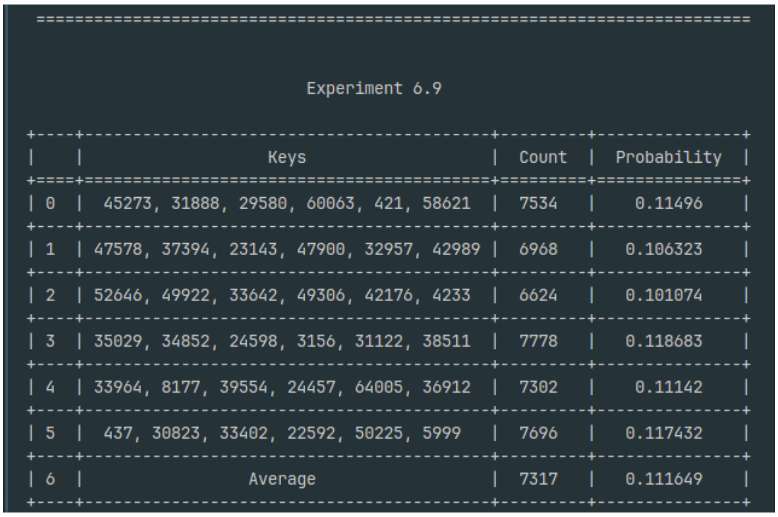
  The function `filterMessage` applies a filtering condition based on the differences between the ciphertexts. The experiment checks which pairs of messages pass the filter after five rounds of encryption.

- **Output**

Keys	Count	Probability
45273, 31888, 29580, 60063, 421, 58621	7534	0.11496
47578, 37394, 23143, 47900, 32957, 42989	6968	0.106323
52646, 49922, 33642, 49306, 42176, 4233	6624	0.101074
35029, 34852, 24598, 3156, 31122, 38511	7778	0.118683
33964, 8177, 39554, 24457, 64005, 36912	7302	0.11142
437, 30823, 33402, 22592, 50225, 5999	7696	0.117432
Average	7317	0.111649

- **Observation** The average number of message pairs that pass the filter is 7317, representing about 11.16% of the total pairs. Compared to Experiment 2, where 5152 pairs followed the target differential, the filtering process retains approximately 70.4% of the useful message pairs.

- **Output Image**



```
==========================================================================


                             Experiment 6.9

+----+-----------------------------------------+---------+--------------+
|    |                  Keys                   |  Count  | Probability  |
+====+=========================================+=========+==============+
| 0  |  45273, 31888, 29580, 60063, 421, 58621 |  7534   |   0.11496    |
+----+-----------------------------------------+---------+--------------+
| 1  | 47578, 37394, 23143, 47900, 32957, 42989 |  6968   |  0.106323    |
+----+-----------------------------------------+---------+--------------+
| 2  | 52646, 49922, 33642, 49306, 42176, 4233 |  6624   |  0.101074    |
+----+-----------------------------------------+---------+--------------+
| 3  | 35029, 34852, 24598, 3156, 31122, 38511 |  7778   |  0.118683    |
+----+-----------------------------------------+---------+--------------+
| 4  | 33964, 8177, 39554, 24457, 64005, 36912 |  7302   |   0.11142    |
+----+-----------------------------------------+---------+--------------+
| 5  |  437, 30823, 33402, 22592, 50225, 5999  |  7696   |  0.117432    |
+----+-----------------------------------------+---------+--------------+
| 6  |                 Average                 |  7317   |  0.111649    |
+----+-----------------------------------------+---------+--------------+
```

Conclusion

The results from all three experiments confirm the theoretical probabilities for differential cryptanalysis on the Sypher004 block cipher. The experiments verify the expected behavior of message pairs after four rounds of encryption, demonstrating how filtering can improve the efficiency of a cryptanalysis attack.