Problem 1:

Brief Explanation:

For this problem, we had to use DES to encrypt a message about Sir Lewis Hamilton and ultimately decrypt the encrypted message to reproduce the message. First, we determined whether the program call used either -e or -d, which identified if the DES was using encryption or decryption. Once the appropriate algorithm is identified, a BitVector object is created with the file pointer created in Python. Thus, 64-bit blocks will be split accordingly with the read_bits_from_file() function. We start analyzing the block if the block has a nonzero size. Furthermore, we account for padding of the BitVectors if the block is not 64 bits long by using the pad_from_right() function. Next, we carry out the Feistel Function by first splitting the 64-bit BitVector into 2 32-bit blocks (RE and LE, for right block and left block). We first expand the right block from 32 bits to 48 bits using expansion permutation. Using the result from this operation, we XOR the result with current round key. The result of the XORing operation is inputted into the substitute function to generate an s-box. Then, we do a permutation using the sbox BitVector object using the p-box permutation table as a parameter. This will help us generated the modified RE, and allow us to set LE to RE and RE to the modified RE. After all rounds are complete, we join the 2 32-bit BitVectors back to a 64-bit BitVector by concatenating RE with LE. Thus, we can write the resulting BitVector in hex format to the output file using a file pointer. This algorithm is the exact same in decryption. The major difference between encryption and decryption is that the round keys will be reversed. Other exceptions include reading the encrypted text file and constructing the BitVector object using the hexstring parameter. Because we are not using the filename when constructing the BitVector, we need to manually index the 64-bit blocks (bitvec = bv[i * 64 : (i + 1) * 64]) to go through each block. After these modifications, we carry out the rest of the operations just like in encryption. The resulting decryption creates a BitVector that will be rendered in ASCII format and written to the decrypted.txt file with the message mentioned earlier.

Encrypted Output:

36d2e582921b6b4a4729ec8a60a4915ba76f3fec1c010014c13444b4afbefb124743582e779a57cf992d87
1fcd7e178fe0c5b2c8ccc1a78fcae1aab4c09dd92388d20af1deaf36212e9fad48d6cf32d8299cf7bfe82e8fa
a32b3383d1877fb86eb489571936cdcda5d32f1bc9a359bd63f411305859fec912107c147cb77b2f459f944
561933e2ca54416929a35c2ce30438568de299dac4a33811a43d6b1e6ec75f86e0768b8ff5eea71a6bb890
7125a17a19997c153b4665123bf24bfe084f129a72292fe22fadf0ab59a06babc93f9aecc82545e35920fa68
a6eea18322458bf5a0fe9e50695326cb0ff211484b883a677b20a3318584f058b818fa594e9bb2744c67a5
ba2ad2d65e39d4522476efa8770e1bf5547cc90f12f73ec93102586e55c8a8e6bdeb8e16205040647bbcb8
be20b29d589da8c3fa2a9ec2f00dc056046c299bbb1532ef8c38b24c021558175055c4a95a1b193deec411
12afa5db015fbac30c6c95c83e3cb07f9b28c849b0330d4b4e84abf996f91ae58a499a44b87340c11ca0074
8b00072d7bf22bb383f3f2e2aa185921e974e23fc695bab5c2ddd27d5fa0e6e6de2af262f2608fa8cbc25bfb
dc4f5f8f0f785a1b4d4c63fa94f0c16601d8cff74856ca0a1ca8e1167db0a5a55e7dbb246202ae59835c16e9
0c1e0c5b2c8ccc1a78f726e8963d971baba5db79b6739f3fa4329acdfef24b1b13d361832c5bd814d7acf70
59e1b251f74e604116ecb90755cc43a12639c01917653cd945c9065737efa9401947fb9557568b567bdf05
9a474f95217f55ba63b3ed666854c2dda688b6acf0722076e3fd18d59b9109d4639c5a10dcc9dd17a3e78f
e956fb9687276ad8aefbfa2764ab669e7444e751fc396940fee2446b2e40d29f277a46ab9781445b25725c
d74215a01694f2/mnt/c/Users/nimal/ECE404/HW2/DES_image.py566b33456851c5966303a2053f6a22d
41581fa810f1668eb7761db9206b466a8a65e50171f030c680a971cffd17e583060cd6e32ec5bd4ba1f9bda
5976a883327bada116974b7e8220290949d5315cd4d308e297b7789bcf7466c433e6effef150ea4a44df49

2f449509044104c47b32351b272672fc599ea6926482920a08dd08cfdfdd19ae50585efebe84f51afbd7487e04b5e127457e37e615da2b55fafc317fecebf59a

Decrypted Output:

In the unforgiving world of Formula One, Lewis Hamilton abides at the top. He's the man to beat, the top earner, the most important voice, the most prominent figure - a Black man alone at the summit of motorsports' highest echelon. England's knight in Mercedes armor. Over the past 15 years, the 36-year-old Briton has won seven world championships, tying the record set by Ferrari's Michael Schumacher - the German F1 driver who was regarded as the greatest of all time until Hamilton broadsided him from that perch. At Sunday's Russian Grand Prix, Hamilton rallied through a late rain shower to claim the checkered flag on the way to becoming the first driver in the sport's history with 100 career victories. And that's besides his 100 career pole positions. As achievements go in racing, this is beyond otherworldly.

Problem 2:

Brief Explanation:

Encrypting an image using DES is like encrypting text files. The major idea in this problem is realizing that the PPM files are in binary format, so we need to read the input image file in binary mode. This also means the encrypted PPM file will be binary format. PPM files have a header that describes the attributes of the file, which take up 3 lines. Thus, we store the 3 lines into a variable and write them immediately after reading them in the output PPM file. Next, we construct a BitVector object using the filename attribute, and carry out the DES encryption algorithm using Feistel structures like what we did in Problem 1. After the rounds have been complete, we write the BitVector object to a file pointer that references to the output file.

Encrypted PPM image: