How to Run:

My AES program should be run from the command line without any parameters. The user will first be asked which mode to run the software in (encryption or decryption). The user will then be prompted to choose file mode or text mode. File mode allows the user to input a filename for the plaintext and key,while text mode allows the user to input hexadecimal text directly. The user is also prompted to choose between cbc mode and ecb mode through a text input prompt.

My implementation will only accept key sizes of 16, 24, and 32 bytes and plaintexts that can be broken up into 16 byte chunks evenly.

Submission Ciphertexts & Plaintext:

```
Aes-ciphertext11.txt:
5d92c6430f49112617f6965dc16c5537d63942be9986f08aac54a1a76a4eee3038999c97307c328
4da3cc7ef614c681394bdb2be10abcef0d6a79040313b3adab92b9330a35468216aa6fea7639515
571306fcd20d6ebc8610196eff15a725f14d17746c6e4a040683234592c2eb8e4e359d2c926e5fe
53bfef8fec01dec599ce552b3648cfa9cfb902f46f0c400ea28738787bbdf16b3964c107d279701
20d8885ab7e0d1b07de669652eefea30c6a1
```

```
aes-ciphertext12.txt:
58e8c8c4ec1d7a6f2d34afe4718a67b855f21c248597ea49b1b7cb6511c96384d07639cc5f1b7b2
fa691c64f3debb4dd69a0f9d94718f45f75b308de7eeea01861f868fbeca354dc70623376908072
3e3bf99978486a8f7278a7343353a8d7ef7001971e8f6bc188bff204cd44f7caf3a3b57d4ef4c30
4fb8e6d2c6554d75b26d2d99f181b8cc687fdd93a462d8691e5f4248ad21bd5d1762ad652ac281d
d8f7239173b85dae4bff8e9ed9b863f560e7d3bed7f7030ea74e81e31be7c38e0aa5
```

```
aes-ciphertext13.txt:
F69cbbcd874ee805dc011eafc91a8632c8e173e7a35aa25b855b5b64ef3654c063f980fc63a2df1
2c14fd562ca8056877e847d67c2130726a1125ac1db2b7e7c22fdf578e3ce819dc531ecc8e6ddd3
8dade2cf3b2ae4160f343f7218f0d5539fc6713cfd09e5d93428f45a0c17a65764e3afed40511cb
8837245e1ee2371ef8b6ec5f868201676668501ca1fd92fa14ac998e61311bdc6def3bd5f18fc03
9aa9d8697f4ff57f95ac5044f8c0a3a2a49162886450aab47b5ec2d96bb82f6864c55a202bd42ec
407c4afd2e2f1c6117e4e
```

```
aes-cbc-ciphertext11.txt (plaintext11 encoded in cbc mode)
5d92c6430f49112617f6965dc16c5537d937381b22ed3e72289d2c2f1bbe5d185dee06ebc5c9725
1e32ec1dfe54b323cc590e114b27572144e8a9d288a2ae39561489866769187829887 64f5905223
2037a868cd53e1763a047d9704857a21d4e186c34dab826fa1b68f03d64be631d4d56faae96e6d9
7817203a34a56539ec06b8d9c58bbbc3be52e34be78dac4b346a2a4a6b6f2e145c6b266fc34bd6f
27c923c90648edb74c28e7bc8a0b6e1b68c7
```

```
Aes-cbc-plaintext10.txt (Decrypted aes-ciphertext10-cbc.txt in cbc mode)
89504e470d0a1a0a0000000d49484452000000a0000000a080200000025058ea0000001949444
15418d36360a002f8ffff3f567126fcda06529a320000d4920308ecf3af170000000049454e44ae
4260820000000000000000000000
```

Entropy

I encrypted two 16-byte plaintexts, one with low entropy and one with high entropy, using the same key and calculated the entropy of each resulting ciphertext:

Key: 8e0ab6172861baef816eba892c52615f

Plaintext 1: 00000000000000000000000000000000
Frequency of “1” = 0
Frequency of “0” = 128
[Entropy of plaintext 1] = 0

Plaintext 2: 2136BA8C76E8081C7D6DDBAD71823BAC
Frequency of “1” = 64
Frequency of “0” = 64
[Entropy of plaintext 2] = 1

Ciphertext 1: 323b68bfcf1cb411a45a49a1ba97762e
Frequency of “1” = 65
Frequency of “0” = 63
[Entropy of ciphertext 1] = 0.99982

Ciphertext 2: c4aeee60b2f1e75ecaec4a58551a1ce4
Frequency of “1” = 65
Frequency of “0” = 63
[Entropy of ciphertext 2] = 0.99982

As we can see from this experiment, the entropy of both ciphertexts is 0.99982 even though the entropy of plaintext 1 is very low and the entropy of plaintext 2 is very high. This shows that the difference in entropy of the plaintexts is not visibly reflected in the entropy of the ciphertexts.

Altering One Bit of Key

Now I will alter one bit of the key. The hex value of the new key is:

Key: 8E**4**AB6172861BAEF816EBA892C52615F

New ciphertext1 (xor) ciphertext1 = 5d4fa3eb8707418339337ea52784bd61

The amount of bits that changed from ciphertext1 to the new ciphertext1 is 65. This is about 51% of the total bits.

New ciphertext2 (xor) ciphertext2 = dc5338763eaec77fb5d779e6b9575076

The amount of bits that changed from ciphertext2 to the new ciphertext2 is 77. This is about 60% of the total bits.

Since changing one single bit of the key changed about 50% of the bits in each ciphertext an attacker would probably not be able to identify which single bit was altered because the ciphertexts with the altered key are extremely different from the ciphertexts with the original key.

Altering Key Length

Now I will add 8 bytes to the key length. The hex value of the new key is:

Key: 8e0ab6172861baef816eba892c52615f**81EF6A7283945B7D**

New ciphertext1 (xor) ciphertext1 = 772bf202aac752920c4aed9d7dffe6f4

The amount of bits that changed from ciphertext1 to the new ciphertext1 is 71. This is about 55% of the total bits.

New ciphertext2 (xor) ciphertext2 = 368fdd312747cbe483007463e3c389f2

The amount of bits that changed from ciphertext2 to the new ciphertext2 is 63. This is about 49% of the total bits.

Since changing the key length to 24 bytes changed about 50% of the bits in each ciphertext an attacker would probably not be able to determine the key length by looking at the difference in each ciphertext because the ciphertexts created using the 24-byte key are extremely different from the ciphertexts created using the original 16-byte key.