Richard Aviles
EECS 565
February 13, 2018

Mini-project 1

I started the project off using lowercase letters. I then tried to switch to uppercase since I didn't know which one was allowed. The following are what I got for my answers for the first 5 codes that were to be brute forced cracked:

1)
Key: em (KS)
Plaintext: CAESARSWIFEMUSTBEABOVESUSPICION
Total time: 0.165627 seconds

2)
Key: dus (JAY)
Plaintext:
FORTUNEWHICHHASAGREATDEALOFPOWERINOTHERMATTERSBUTESPECIALLYINWAR
CANBRINGABOUTGREATCHANGESINASITUATIONTHROUGHVERYSLIGHTFORCES
Total time: 6.04341 seconds

3)
Key: cqex (IWKD)
Plaintext: EXPERIENCEISTHETEACHEROFALLTHINGS
Total time: 139.456 seconds

4)
Key: tcyhz (ZIENF)
Plaintext: IMAGINATIONISMOREIMPORTANTTHANKNOWLEDGE
Total time: 2833.11 seconds

5)
Key: buweyl (HACKER)
Plaintext:
EDUCATIONISWHATREMAINSAFTERONEHASFORGOTTENWHATONEHASLEARNEDINSC
HOOL
Total time: ?? seconds (I accidentally exited the terminal and did not get to save the time)

For the project, I tried to make the password cracker more efficient by separating my dictionary into lengths so that I would not have to search the entire dictionary every time I needed to compare. I created a Node class that held the data and linked list the data using an array to hold the linked list of each word size. I also converted everything to lowercase/uppercase to save time on having to distinguish between upper and lowercase. Doing all this made it possible to crack all the codes in a somewhat reasonable time.