**Appendix:**

**modifiedurl.py**

import urllib.request

f = open('FemaleNames.txt', 'w')

local\_filename, headers = urllib.request.urlretrieve('https://cs.brynmawr.edu/Courses/cs330/spring2018/FemaleNames.\

txt')

html = open(local\_filename)

line = html.readline()

while line:

#print(line)

f.write(line)

line = html.readline()

f.close()

**SoundexOnFile.py**

import Soundex

f = open("FemaleNamesSortedUnique.txt", "r")

f2 = open("FemaleNamesMatched.txt", "w")

line = f.readline()

while line:

lineSoundex = Soundex.soundexNaive(line.upper(), len(line))

f2.write(lineSoundex + ' ' + line)

line = f.readline()

f.close()

f2.close()

**squash.py**

f = open("FemaleNamesSortedMatchedSoundex.txt", "r")

f2 = open("FemaleNamesSortedMatchedSoundexCompressed.txt", "w")

line1 = f.readline()

line2 = f.readline()

list1 = line1.split()

list2 = line2.split()

f2.write(list1[1] + ' ')

while line1:

if len(list1) > 0 and len(list2) > 0 and line1 and list2:

while len(list1) > 0 and len(list2) > 0 and list1[0] == list2[0]:

f2.write(list2[1] + ' ')

line2 = f.readline()

list2 = line2.split()

# get a new line, since the list value is no longer the same

line1 = line2

list1 = list2

f2.write('\n')

else:

break

ADD METAPHONE HERE

**How programs were used in exercise:**

1. Got FemaleNames.txt by using modifiedurl.py to read website contents into a .txt file
2. Ran a tr -d [:punct:] < FemaleNames.txt |tr A-Z a-z|sort|uniq >FemaleNamesSortedUnique.txt command to get FemaleNamesSortedUnique.txt which had all names as unique and as lower case
3. Ran SoundexOnFile.py and MetaphoneOnFile.py, respectively, to get the signature name pairings of the encoding to the name. This was output into FemaleNamesMatchedSoundex.txt and FemaleNamesMatchedMetaphone.txt, respectively.
4. I then sorted the resulting .txt file. sort FemaleNamesMatchedSoundex.txt > FemaleNamesSortedMatchedSoundex.txt. The equivalent for Metaphone, too.
5. Then I ran squash.py for each text file to create space-separated lines of names which sound alike for each algorithm.
6. Finally, I ran the awk -f filter.awk outputFromStep5 > soundexClasses.txt and MetaphoneClasses.txt, respectively. This is what gave me the information.
7. Finally, I wrote a short analyze.py file that looked at the contents of a file and printed out the sought information.