Cameron Morales
Artificial Intelligence 1
Assignment 4

My implementation of this name generator utilized a Second-Order Markov model which utilizes 2 letter sequences as determinants of subsequent letters, yielding the following associations: 'xy' -> ‘z’, where x, y and z are three contiguous letters in a given name. I managed to receive generally appropriate results utilizing this model; example runs utilizing different parameters are displayed below:

Males

```
****** Baby Name Generator ******
---------------------------------

Type 0 for Females, 1 for Males:1
Enter the minimum length:4
Enter the maximum length:10
Enter the number of names to be generated: 30

---------------------------------

Alawn
Ryah
Zave
Tras
Darick
Holand
Eauriovan
Elson
Krip
Kady
Caldeas
Aaddine
Kayvon
Armestin
Broctaylon
Madiel
Kyahaad
Cayleonn
Izaciah
Urtus
Wayleace
Kiel
Sagge
Judy
Madyn
Ston
Kevon
Nael
Sane
Alandense
```

Females

```
****** Baby Name Generator ******
---------------------------------

Type 0 for Females, 1 for Males:0
Enter the minimum length:4
Enter the maximum length:10
Enter the number of names to be generated: 30

---------------------------------

Emmaly
Ryah
Marah
Grey
Janzann
Jayte
Matha
Khlyson
Ania
Gwen
Karie
Briny
Ganne
Kareliel
Alda
Renna
Hanian
Kyelie
Colice
Seriss
Ashelinge
Harlillen
Adela
Vireth
Janney
Maylah
Bianne
Damryah
Marbaina
Shanna
```

A subjective analysis of these results reveals the importance of the length parameters in producing functional and aesthetically pleasing names. Functionality is clearly limited by the min and max lengths if we assume that a name's usability is dictated by its syllable count and consonant-vowel combinations. Name length also correlates with aesthetics in terms of the increasingly foreign-sounding nature of the results. Within this context you can observe the limitations of the second-order model in producing names that appropriately resemble the reference items utilized during its construction. The state selection process in this model is only limited by the most recent state, leading to an amalgamation of consonant-vowel combinations which cause the foreign-sounding nature of the names. Regardless, the model constructed is robust enough to produce some appropriate letter sequences when given practical input lengths. Example parameters for which I found modest success are displayed in the screenshots above.

My sentiments regarding the success of the model to produce appropriate names under the context of gender orientation generally mirror the observations regarding the length parameters. At shorter lengths, we can find names that reflect slight derivations, or misspellings from more common names, such as ‘Darick’ and ‘Nael’ for men, or ‘Shanna’ and ‘Karie’ for women. At longer lengths, the gender orientation of the names often hold up, albeit under a significant degree of relaxed scrutiny. Purportedly male names such as ‘Dardolden’, or female names such as ‘Yanatashaya’, still maintain some characteristics, such as recurring subsequences, that could differentiate their gender associations despite exhibiting pronounced aesthetic deviation into randomness or foreign-ness.

There are several approaches that might bolster this program’s ability to produce aesthetically appropriate names. We could write domain-specific rules to be enforced during model construction or word generation. For example, we could bias certain vowel and consonant combinations, or certain name termination sequences. Additionally, and perhaps most importantly, we could modify this program to operate as an N-th Order Markov Model. This could limit the randomness that is especially rampant at greater lengths by biasing name construction towards more oft repeated n-length subsequences visible in the reference names. I integrated this variable length chain concept into my program as a user provided argument. Example runs utilizing this new input parameter are displayed below:

Males

```
****** Baby Name Generator ******
--------------------------------

Type 0 for Females, 1 for Males:1
Enter the minimum length:6
Enter the maximum length:15
Enter the number of names to be generated: 10
Enter the order of the model (between 2 and 5):4

--------------------------------

Dashad
Benjamison
Jonathanial
Noland
Jonathen
Tuckery
Rashawn
Jadonis
Rodericky
Abrahim
```

Females

```
****** Baby Name Generator ******
--------------------------------

Type 0 for Females, 1 for Males:0
Enter the minimum length:6
Enter the maximum length:15
Enter the number of names to be generated: 10
Enter the order of the model (between 2 and 5):4

--------------------------------

Adisyn
Destin
Karmani
Loreleine
Heatherine
Atheresa
Christiny
Savanahi
Clarisol
Eleanna
```