PRESIDENT PARSER MANUAL

John Flahive, Daniel Stekol

1. General

- How it works: First, the Web Crawler retrieves the presidential speeches for a particular president from millercenter.org. Then, once the speeches are retrieved as strings, they are passed one by one to the Parser, which interacts with the Word class to compile a directed graph representing the speech, where each node is a word in the speech and each edge (u -> v) represents the fact that word u is followed by word v somewhere in the text, and the edge weight represents the number of times that this word pair appears. Additionally, each incoming edge is linked to an outgoing edge: for instance, if the phrase "u v w" appears in the text, then word v would have an incoming edge form u and an outgoing edge to w, and the edge (u -> v) would be linked to the edge (v -> w) since that is the path that the actual speech followed.
- How to run: To run this program, Java and the external library JSoup are required. Given these requirements, the program can be run in an IDE from the Driver.java file. Running the program will present the user with the President Parser. Within the President Parser, there are 3 tools. Entering in a given, described character brings the user to each different tool. Entering in 'G' or 'g' will bring the user to the Phrase Generator tool. Entering in 'M' or 'm' will bring the user to the Phrase Matcher tool. Entering in 'Q' or 'q' will quit from the President Parser program.

```
-- President Parser --
(Phrase Generator) : type G
(Phrase Matcher) : type M
(Quick Info) : type I
(Quit) : type Q
```

2. Phrase Generator

- How it works: for each word, the program randomly generates the next word based on the current word's out-neighbors. The edge weights are used to weight the probability distribution toward word sequences that are more commonly encountered.
- How to run: First, enter in a president's name. Wait until the program is finished parsing this president. Note that the entered string will be matched to a president containing that string, so entering 'Clint' will match President Clinton. This president will be used to generate phrasing. Then, either enter 'G' and a phrase length to generate a phrase or enter 'Q' to quit to main menu.

```
-- Phrase Generator --
Enter a president's name:
```

3. President Matcher

- How it works: for each president, the program parses that president's speech graph – then, it finds the shortest path containing the word sequence provided (using BFS, and therefore not accounting for edge weights). The longer this path length, the less likely this string is to appear in that president's speeches, since a high path length means those words to not appear close to each other in the actual speeches. These scores can then be used to rank the presidents' similarity: the president with the lowest path length is the best match. This method is essentially an alternative to the tf-idf method of query matching. Note: the BFS code is adapted from a CIS 121 homework assignment.
- How to run: First, enter in a sentence to match to presidents. Then, enter in a president's name to include this president in the matching program or enter 'C' to calculate with all the current included presidents. When entering a president's name, make sure to wait until the program finishes parsing that president. After choosing to parse a president, another president can be parsed, or the match can be calculated. After calculating, the user will be presented with an option to run the matcher again, or to quit to the main menu.

```
-- Phrase Matcher --
(Run Matcher) : type M
(Quit to Main Menu) : type Q
```