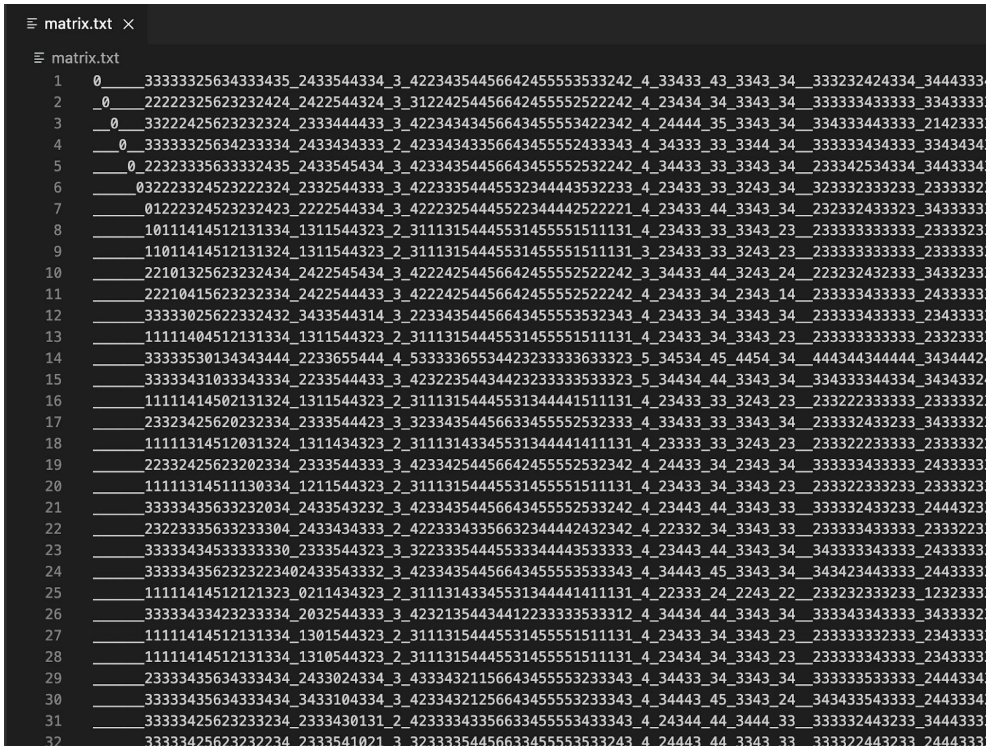
Results

We created a replica of the Wikispeedia game by loading in the list of pages and connecting the links between pages. One of the main aspects of our project included generating an adjacency matrix for the graph of the pages and their edges. Using Floyd Warshall's algorithm, we created the adjacency matrix and stored the shortest distances between pages in the matrix.txt file to avoid calling the algorithm repeatedly. Here is a small sample of the full matrix:

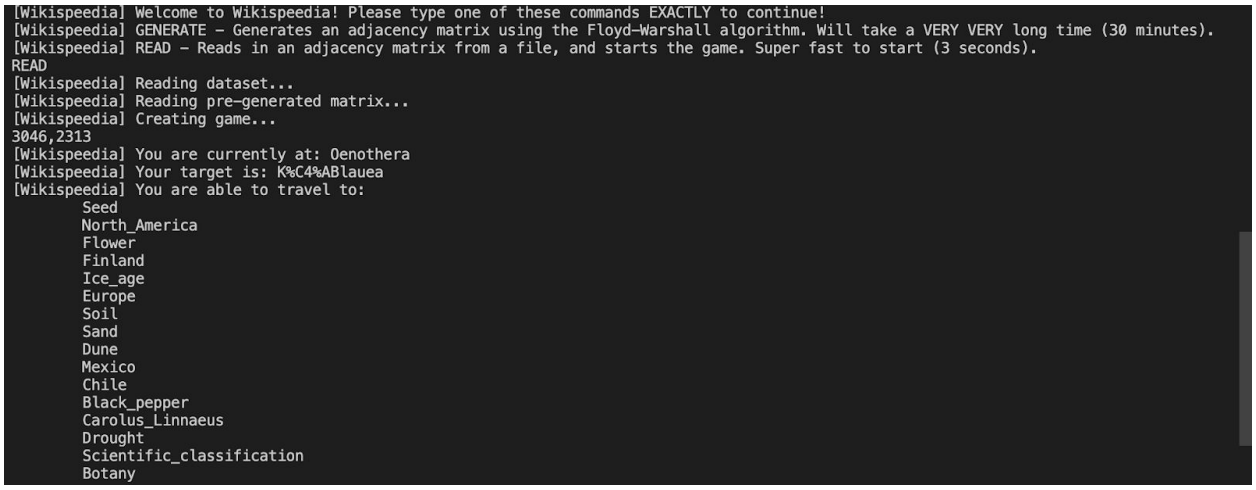
The "_" indicates that there is no path from a Node at the row index to the Node at the column index. We discovered it is much faster to run the Floyd Warshall algorithm only once and then use this resulting matrix to run future games.



Our resulting Wikispeedia game allows the user to generate the adjacency matrix with the GENERATE command. This will utilize the Floyd Warshall algorithm, which took 1-2 hours (depends on computer) to complete on our default dataset of 4600 pages. We discovered that the O(n^3) runtime of the Floyd Warshall algorithm would make it hectic to run repeatedly.

The READ command reads in an already provided adjacency matrix from the provided file path to generate the shortest distances between pages, which is much faster than generating.

This below is the result of our game, which includes the current location a player starts at, the target location, and the list of pages a user can visit from the current page. There is also the option of typing in “exit” to leave the game at any time.



The path taken by the player will be printed in the console, and the most optimal path will be shown using the Iterative Deepening DFS algorithm even if the player gives up. This involves the shortest path from the starting page to the ending page.