Dataset: English Speaking Twitch users and friends (Nodes: 7,126, Edges: 35,324)
Link: https://snap.stanford.edu/data/twitch-social-networks.html

The dataset I chose is an undirected dataset that describes Twitch social networks. Twitch is a live streaming platform where thousands of creators can stream games, live moments, and much more to audiences of viewers. This dataset consists of six folders where the dataset is split by users' language. I decided to only look at the English speaking users and the relationships between them. In this dataset, there were 7,126 nodes and 35,324 edges where the nodes represent individual users. Since this dataset only looks at the English speaking users, this dataset is relatively small compared to other social platforms.

I have had some experience with using Twitch in the past and thought it would be an interesting dataset to analyze. The main form of analysis I will be using is Breadth First Search to help analyze the connectivity of the dataset and use other auxiliary functions to determine the average distance between nodes and other insightful data.

I used three separate modules in addition to the main file in order to complete this project. The first module is where I defined the create_adjacency_list function and another function that just prints the adjacency list. The adjacency list is used in the Breadth First Search function. The second module is where I made all of the search functions. The first of the search functions performs a Breadth First Search for every node in the dataset. It then returns a vector of vectors with the distances between every node. Since my dataset is relatively small, running this function only takes a 1 minute to run. I also have a separate function that prints out all of the BFS results for each individual node. The last search function I have computes BFS for just one selected node up to a certain number of nodes selected. Since this function is just looking at one node, its runtime is very fast and it can be used on nodes of interest.

The last module contains all of the graph analysis functions I thought would be relevant to this dataset and methods associated with it. The first function calculated the average distance between all nodes. When I used Twitch in the past, I did not have many followers as Twitch is mainly a site to engage with creators and not other viewers. Due to this, I guessed the average degrees of separation would be high. The average ended up being 3.677 which was a little lower than I had initially expected. The next function returned the pairs of nodes that were the furthest away from each other in the dataset. The furthest nodes in this particular dataset were 10 degrees of separation away from each other. The last function calculates the percentage of connections that occur at different degrees. This distribution was interesting to look at and the most common degree of separation ended up being 4 with 42.94%.

Outputs:

```
Distances from BFS for node 15 (from 0 up to node 100): 0:5 1:3 2:3 3:4 4:4 5:2 6:3 7:4 8:4 9:2 10:4 11:4
12:4 13:3 14:4 15:0 16:4 17:4 18:3 19:4 20:4 21:4 22:3 23:3 24:3 25:4 26:3 27:4 28:4 29:4 30:3 31:4 32:3 3
3:4 34:4 35:4 36:3 37:3 38:3 39:4 40:2 41:4 42:4 43:4 44:4 45:4 46:2 47:4 48:4 49:4 50:5 51:4 52:4 53:4 54
:3 55:3 56:3 57:3 58:3 59:4 60:4 61:3 62:3 63:4 64:3 65:3 66:2 67:5 68:4 69:4 70:4 71:5 72:4 73:4 74:3 75:
4 76:4 77:3 78:4 79:4 80:3 81:3 82:4 83:4 84:4 85:3 86:4 87:3 88:3 89:3 90:3 91:3 92:5 93:2 94:3 95:3 96:3
 97:4 98:4 99:3 100:3 The average distance is 3.677099644749034
```

This is the output of the One BFS function where the input node was 15 (from node 0 - 100). When looking at the One BFS function output like this, the average distance of 3.677 makes more sense as most of the distances here lie around 3 or 4.

```
The average distance is 3.677099644749034
Nodes with the maximum distance are
node 241, node 2608 with distance 10
node 241, node 6012 with distance 10
node 2608, node 241 with distance 10
node 2608, node 3981 with distance 10
node 2608, node 4735 with distance 10
node 3981, node 2608 with distance 10
node 4735, node 2608 with distance 10
node 6012, node 241 with distance 10
Percentage of nodes at distance 1: 0.14%
Percentage of nodes at distance 2: 6.22%
Percentage of nodes at distance 3: 36.15%
Percentage of nodes at distance 4: 42.94%
Percentage of nodes at distance 5: 12.53%
Percentage of nodes at distance 6: 1.82%
Percentage of nodes at distance 7: 0.19%
```

This is the output of the average distance, furthest nodes, and degree distribution functions.

Test Output:

```
running 2 tests
test avgdistancetest ... ok
test furthesttest ... ok

test result: ok. 2 passed; 0 failed; 0 ignored; 0 measured; 0 filtered out; finished in 0.00s
```

I created two tests that checked the accuracy of the furthest node and average distance functions. For both of these tests, I used the same random dataset and calculated the average distance between the nodes and the furthest nodes by hand. Both of these tests passed.