Data Structures

For my program I have used the following main data structures:

ADS2Graph

---------

* I use a double 2D array as an adjacency matrix to identify which pairs are friends.
* I use a boolean array to identify which items have been visited by an algorithm.
* I use a double array to hold tentativeDistances within the Shortest Path Algorithm.

ADS2List

---------

* I used a simple String array to hold the full names and idenify them using the index.

(1: Belkis Pileggi is index 0 in the array).

# Algorithms

For my program I have used the following algorithms:

Search Algorithm : FindUserID

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

This algorithm finds the UserID by compaing the provided string "fullName" to

each index within the ADS2List array. When it finds the user it returns the UserID.

Shortest Path Algorithm : FindPath

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

I use this algorithm to find the shortest path between startID and destination.

It overwrites the tentativeDistance[i] with the shortest distance possible.

Sort Algorithm : SortAlgo

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

While the file is not sorted the algorithm loops through the for loop checking the current index

value 'allPath[1][i]' against the next indexs value. If the value of the first is higher than the next,

they are both swapped and the sort continues.

When this loop is completed without any changes being made, the counter will remain 0 and

the recommended String[] will be created and returned.

I created this algorithm off the top of my head so I'm unsure if it fits a standard convention.

It looks like this is a bubble sort algo or very similar.

Although I haven't been able to complete much testing yet, I would assume this algorithm is inefficient and can be improved upon. Bubble sorts are considered to be an inefficient way of sorting for large pools of data, however becuase the size of this data is small it is not a huge problem yet.

There are plenty more efficient algorithms I could implement. An insertion or selection sort may be better, however a Heap Sort algorithm should be much more efficient and would be my chosen implementation.