CS2302 Data Structures

Fall 2019

Lab Report #6

Due: 1 November 2019

Professor: Olac Fuentes

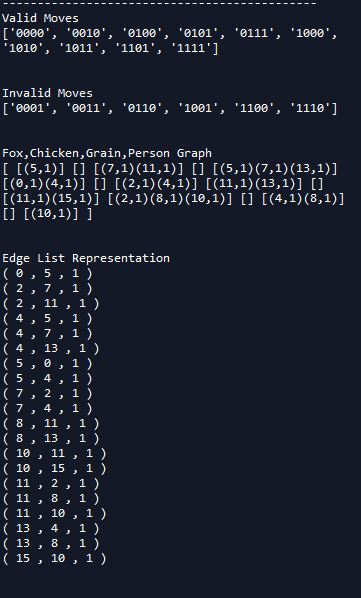
TA: Amindita Nath

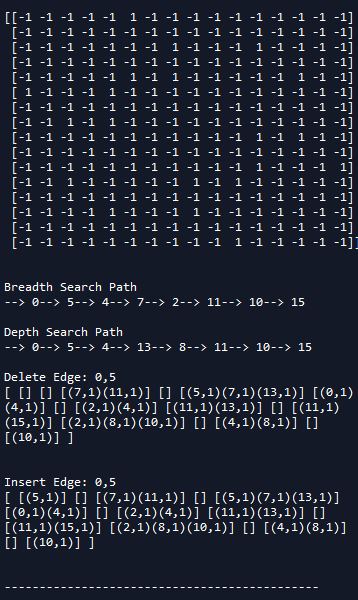
**Introduction**

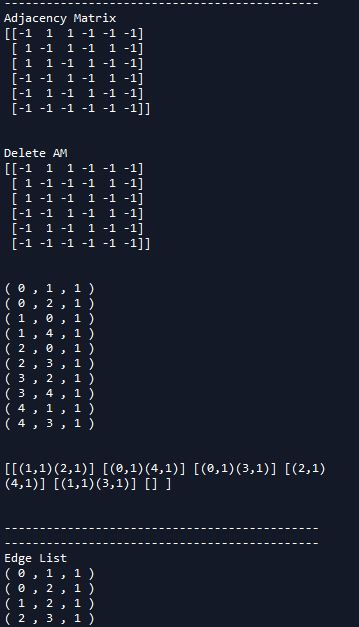
The problem presented in this lab is relating to the riddle of crossing a river with a fox, chicken, grain, and a person by implementing a graph and search functions to establish a path to the solution where all the fox, chicken, grain, and a person cross the river within the limitations.

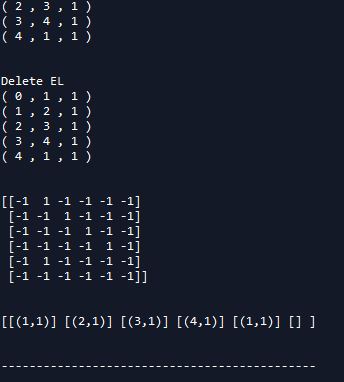
**Proposed Solution Design and Implementation**

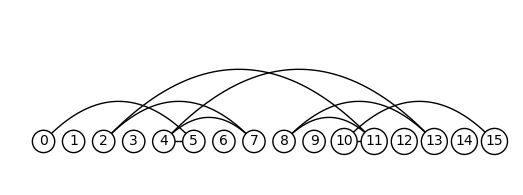
Design –











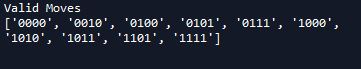
Operation #1: The first operation I separated into segments and implemented insert edge, delete edge, and display for each graph representation as well as add representation for the three types of graphs for each type of graph.

Operation #2: To implement the graph to solve the riddle I first created a list of 4 bit binary from 1 – 15 and cut down the list by checking if the fox and chicken are left together alone or if the chicken and grain are left alone together. Then after cutting down the list I added the edges into the graph to create the graph and implemented both Breadth-first search and Depth-first search to find a path to solve the riddle.

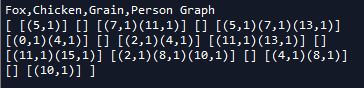
**Experimental Results**

The running time for the breadth first search and depth first search is O(V+E).

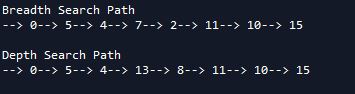
The search methods got though all of the vertexes and edges leading to this running time. I first began by creating a list legal moves in binary



After creating valid move list created edges by checking if a move was a 1- or 2-bit change and if the person was a different number then the next move because the chicken, fox, and grain cannot move without the person moving.



I represented the binary strings as integers so it is easier to visualize, each binary digit added to the graph corresponds to the index number so 0010 represents the second index. I then sent the graph that was made to a breadth-first and depth-first function to create a path for the solution.

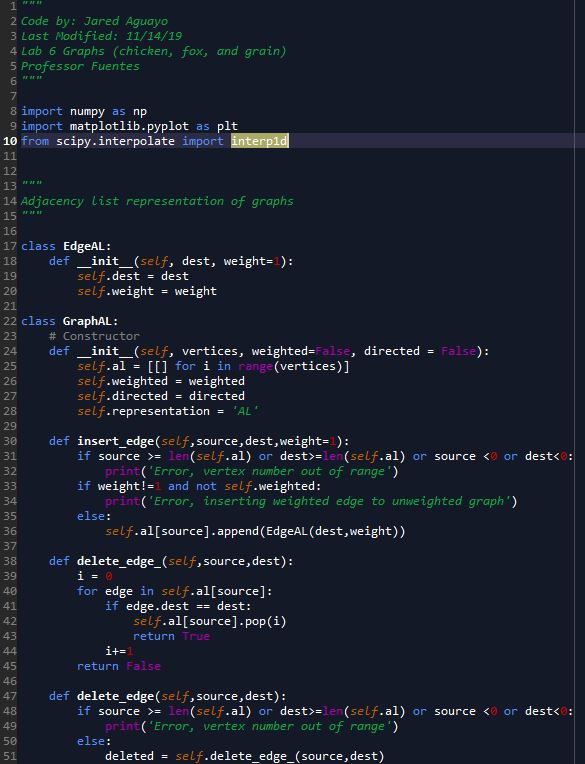


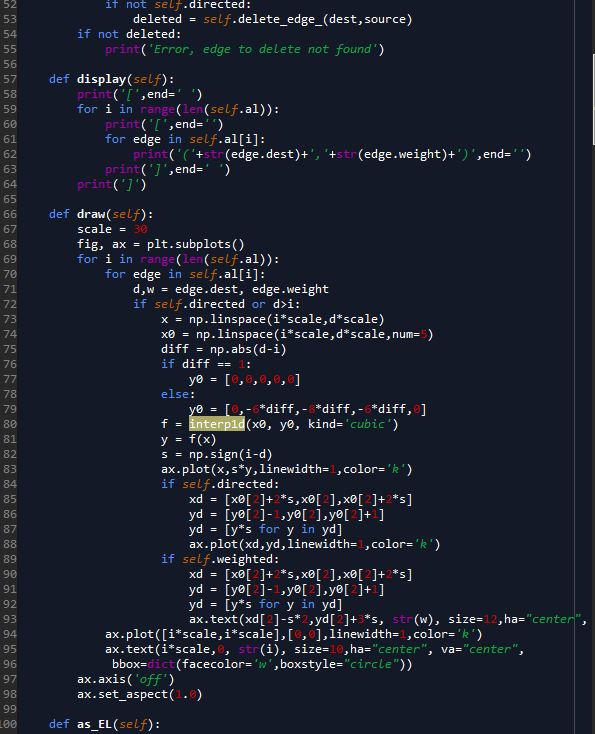
The output is printed out by using the path list that I used in each of the search functions which shows these two paths.

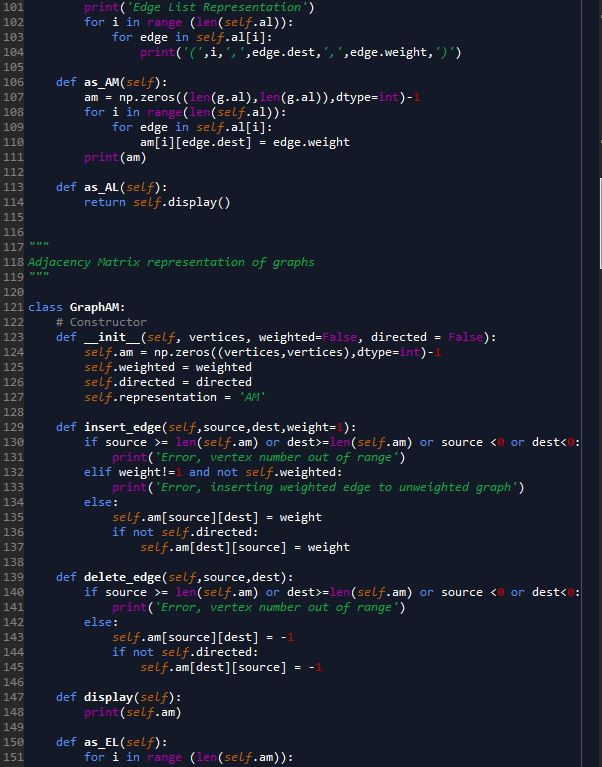
**Conclusion**

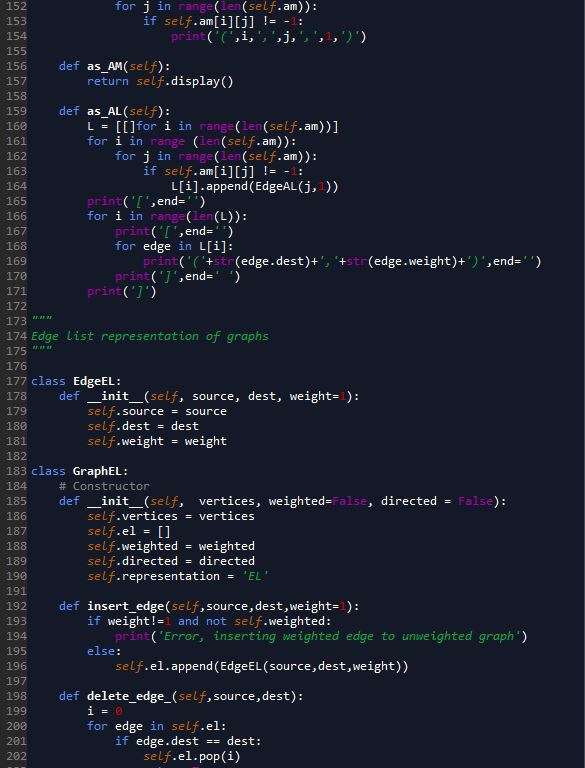
I learned how to use graphs to find paths, which is important because it is similar to what a program does when giving directions to a destination. I also learned how to better understand graphs as in understanding the difference between vertices and edges and how they are both used to find paths in graph.

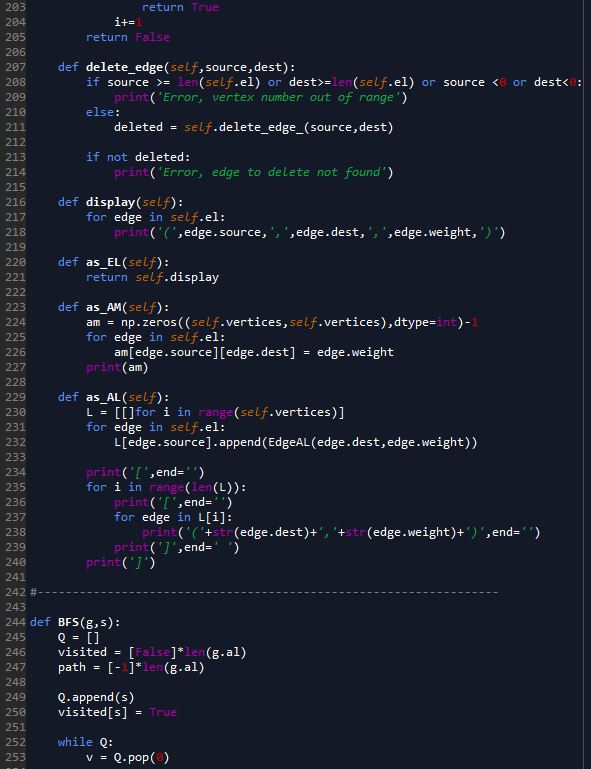
**Appendix**



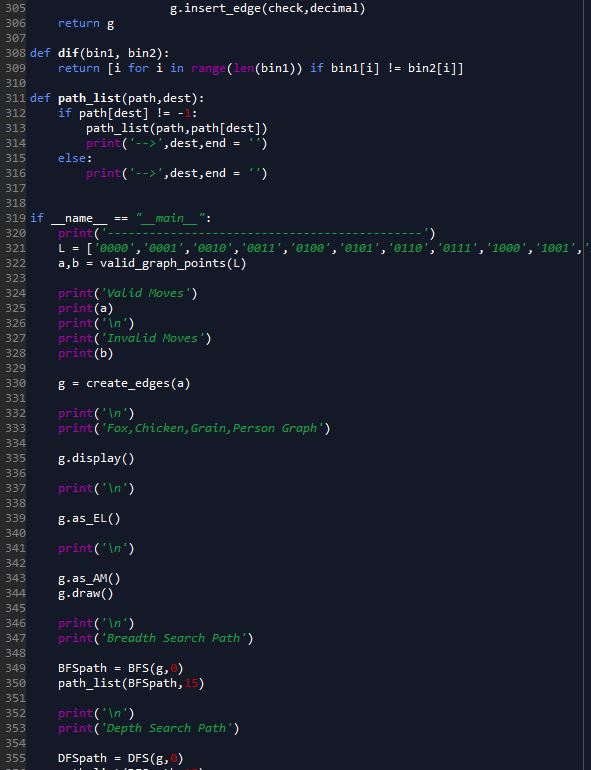


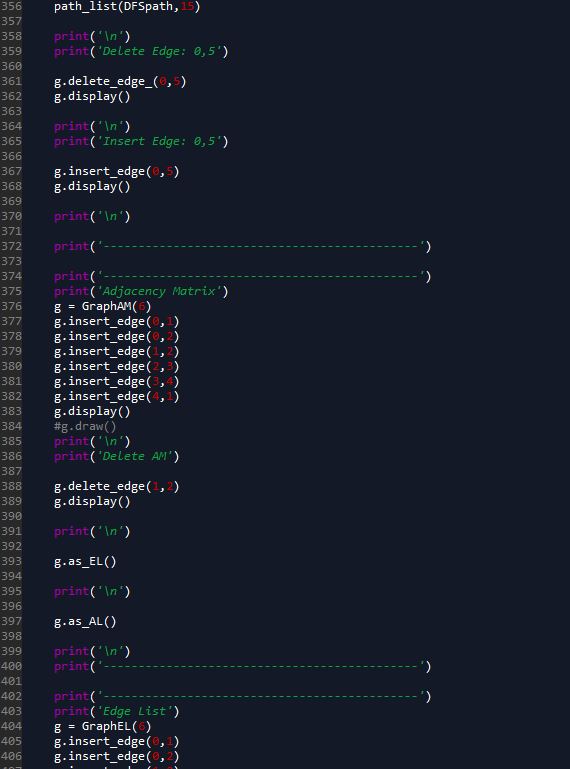


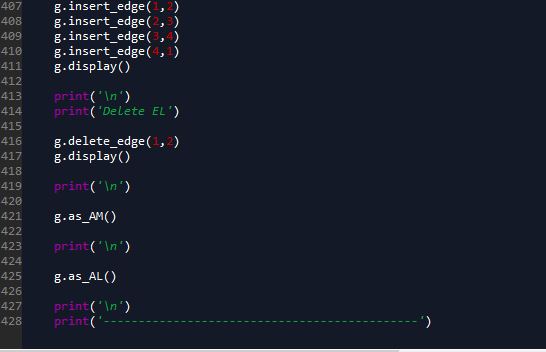












I certify that this project is entirely my own work. I wrote, debugged, and tested the code being presented, performed the experiments, and wrote the report. I also certify that I did not share my code or report or provided inappropriate assistance to any student in the class