

CSIS385 - Academic Showcase Project

Names: Andrew Brockley, Andrew Jop, Chris LaDuke, Fred DeKoker

Proposal:

For our project, the METAL data aspect will study Tarjan's Bridge Detection algorithm. This algorithm detects the edges in a graph that serve as the only path to some subset of points within a graph. The application of this algorithm in reality could be in the field of emergency preparedness/response. In the anticipation of a big storm, utility/fire responders may be curious what roads should be of the most concern in their city. In this context, bridges will be of the most concern because if that road were to wash out or have a tree fall, some percentage of residents will be left isolated from the rest of the city. So, we will apply Tarjan's Bridge Detection algorithm to a graph consisting of roads and intersections to various storm-prone cities. The second piece of our project will be a code-based implementation of the game show Wheel of Fortune. This code will randomly select a phrase from a text file, produce a random number as the each player's 'spin', and track money totals for each player until the phrase is completed. This code will utilize brute force string search methods to see if a player's letter guess or phrase guess is correct. Progress updates will be provided on: 3/22, 3/29, and 4/12.

Update 1 (3/22/2024):

So far, our group has analyzed the pseudocode on the METAL HDX site for Tarjan's Bridge Detection algorithm as well as familiarized ourselves with brute force string matching algorithm best and worst cases. We have also looked at the implementation of code that loops for user input that will be essential to our wheel of fortune game. This will allow us to implement both of these algorithms using some sample data in the coming days to ensure that we have working code so we can begin expanding the scope of our project to the full capabilities we seek.

Update 2 (4/05/2024):

We have worked out the design of our Tarjan's Bridge detection, to do this we are going to pull similarities in HighwayGraph.java from lab 2. This has the structure and methods of handling vertices and edges within a graph structure. We have also been working through how we want to implement our version of Jeopardy, we are going to have players with \$ amounts and arrays that will hold all alphabetic characters, as well as vowels. After the user enters a guess, we will use brute force string matching to verify the correctness of the answer made against the phrase array.