University of Nevada, Las Vegas Computer Science 302 Spring 2017 Assignment 4: Due September 28, 2018 Using a Queue to Solve the Minimum Path Problem

You are permitted to work in groups, get help from others, read books, and use the internet. You may not copy code electronically, nor share your code electronically with any other student.

- 1. Write a program which uses a circular queue to solve the single source shortest path problem for a graph.
- 2. Submit to the graduate assistant, Shradha Kapoor, by midnight Friday September 28.
- 3. Use the circular linked list implementation of a queue to implement breadth first search on a graph G.
 - (a) G is illustrated below.



- (b) Use 43 as the start vertex, s.
- (c) The output of the algorithm consists of two arrays. If x is a vertex, then d[x] = the distance from s to x, while back[x] is the next to the last vertex on a shortest path from s to x.

- (d) The edges of G are listed at http://web.cs.unlv.edu/larmore/Courses/CSC269/biggraph.txt
- 4. (a) Use the circular linked list implementation of queue.
 - (b) Run the algorithm twice: once with the O(1) time implementation of enqueu, the other with the linear time implementation.
 - (c) Use a variable kounter to estimate the time complexity of each version of your algorithm.
 - (d) Using the array of back pointers, compute and print out the shortest path from the start vertex to some other vertex, of your choice, which is not close to the start vertex.
- 5. Here is a small instance of the problem.



- 6. The list of edges of for this small instance is
 - 86
 - 69
 - 94
 - 19
 - 1 2
 - 52
 - 4 6
 - 58

7. The list of edges of the small graph is at: http://web.cs.unlv.edu/larmore/Courses/CSC269/smallgraph.txt

8. The output for this small instance should look like this:

1 3 2 2 2 5 3 4 2 6 5 1 8 6 1 8 7 8 0 9 2 6 kounter = The shortest path from 8 to 9 is 8 6 9