CMPS 1500 Introduction to Computer Science I – Fall 13

11/12/13

10. Homework

Programming portion due **11/21/13** at 11:55pm on Blackboard.
Written portion (problem 3) due **11/22/13** at the beginning of class.

In order to receive any credit for the programming portions, you are required to thoroughly comment and test your code.

1. **Diameter (6 points)**
   The *diameter* of a graph G is the maximum over all shortest path distances between any pair of vertices u, v in G. Write a Python function that computes the diameter of graph G, given as a dictionary of adjacency lists. Test your code (also with the graph in problem 3 below).

2. **Connected Components (6 points)**
   Given a graph G as a dictionary of adjacency lists. Write a Python function that returns the number of connected components in G, and that for each connected component prints a line listing its vertices. Test your code (also with the graph in problem 3 below).
   (Hint: Extend the breadth-first search code that we covered in class, such that it repeats for one start vertex from each connected component.)

3. **Degree of Separation (8 points)**
   Consider the following scenario, where friendship is mutual (i.e., if A is friends with B, then B is also friends with A): Charlie is friends with Anna, Bob, and Don. Anna and Bob are also friends. Don is friends with Emma, Frank, Charlie, and Greg. Emma, Frank, and Isaac are friends. Hannah, James, and Greg are friends.

   (a) (4 points) Model this scenario as a graph, and draw the graph. How many vertices and edges does it have? How many connected components does it have?

   (b) (2 points) What is the largest degree of "friendship separation" between any two people in this scenario? How could you compute this?

   (c) (2 points) Model this graph using a dictionary of adjacency lists, and use it as a test case for problems 1 and 2 above.