ASSIGNMENT 6

DUE: Wednesday November 1, 7 PM. DO NOT COPY. ACKNOWLEDGE YOUR SOURCES.

Please read `http://www.student.cs.uwaterloo.ca/~cs341` for general instructions and policies.

1. [10 marks] **2-neighbours.** For a vertex v in an undirected graph $G = (V, E)$, the *degree* of v, denoted $d(v)$ is the number of neighbours of v, i.e., $|\{u \in V : (v, u) \in E\}|$.

   Define $s(v) = \sum \{d(u) : (v, u) \in E\}$. Define $d_2(v) = |\{w \in V : \exists u \in V \text{ s.t. } (v, u) \in E \text{ and } (u, w) \in E\}$.

   (a) [2 marks] Give an example to show that $s(v)$ can be different from $d_2(v)$.

   (b) [4 marks] Give a linear time algorithm to compute $s(v)$ for all vertices v. (Assume that the graph is given as adjacency lists and that addition costs $O(1)$.)

   (c) [4 marks] Give an $O(n^3)$ time algorithm to compute $d_2(v)$ for all vertices v.

2. [5 marks] **Reconfiguration graphs.**

   (a) [3 marks] Consider the goat, wolf and cabbage problem. You are on one side of a river with your goat, your cabbage, and your wolf. (`https://xkcd.com/1134/`)

   You have a boat but it only fits you and one other. The wolf will eat the goat unless you are there to stop it. Also, the goat will eat the cabbage unless you are there to stop it. How can you transport everyone across the river?

   Make a directed graph to capture the possibilities. Vertices correspond to configurations of who is on each side of the river, e.g. $[gcwY|\phi]$ is the initial vertex, where g is the goat, Y is you, etc. , and $[cw|gY]$ means you and the goat are on the other side of the river. A directed edge (c, c') means that one boat trip can change configuration c to configuration c'. There would be $16 = 2^4$ vertices except that 6 of them are ruled out because of the wolf eating the goat or the goat eating the cabbage.

   Draw the 10 vertex graph with directed edges. How many directed edges are there? What graph problem do we need to solve in order to answer the question of how to transport everything across the river?

   (b) [2 marks] Consider the Rubik's cube problem: you are given a configuration of the Rubik's cube (a cube mixed up) and a number k and you want to know if this configuration can be solved with at most k moves. Suppose we apply the same approach as above, i.e., create a graph whose vertices correspond to the configurations of the Rubik's cube and whose edges correspond to moves. Let v_0 be the vertex corresponding to the initial mixed up cube and let v_f correspond to the solved configuration of the Rubik's cube. What graph question gives us the answer we want? Is this a practical solution?