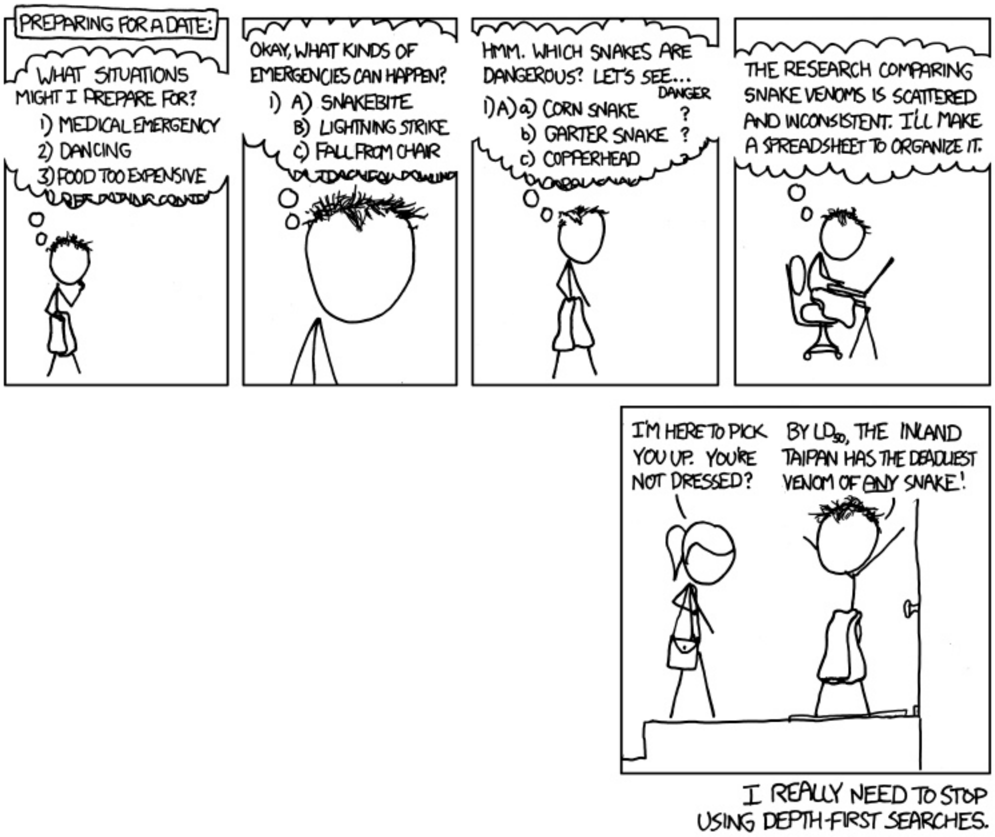
CS302 - Assignment 16

Due: Tuesday, April 23 at the beginning of class
Hand-in method: paper


PREPARING FOR A DATE:
WHAT SITUATIONS MIGHT I PREPARE FOR?
1) MEDICAL EMERGENCY
2) DANCING
3) FOOD TOO EXPENSIVE
OKAY, WHAT KINDS OF EMERGENCIES CAN HAPPEN?
1) A) SNAKEBITE
B) LIGHTNING STRIKE
C) FALL FROM CHAIR
HMM. WHICH SNAKES ARE DANGEROUS? LET'S SEE...
DANGER
1)A) a) CORN SNAKE ?
b) GARTER SNAKE ?
c) COPPERHEAD
THE RESEARCH COMPARING SNAKE VENOMS IS SCATTERED AND INCONSISTENT. I'LL MAKE A SPREADSHEET TO ORGANIZE IT.
I'M HERE TO PICK YOU UP. YOU'RE NOT DRESSED?
BY LD50, THE INLAND TAIPAN HAS THE DEADLIEST VENOM OF ANY SNAKE!
I REALLY NEED TO STOP USING DEPTH-FIRST SEARCHES.

http://xkcd.com/761/

Notes:

- Many of the algorithms below can be accomplished by either modifying the graph and applying a known algorithm or slightly modifying a known algorithm. Try thinking of these *first* as they will save you a lot of work, and writing :)
- You will be graded on efficiency!
- If not specified in the problem, you may assume whatever graph representation makes your algorithm more efficient (adjacency list or adjacency matrix). State which one you are using.

1. **[5 points] Induction on Graphs**

   Use induction to prove that any connected, undirected graph $G = (V, E)$ satisfies $|E| \geq |V| - 1$.

2. [**5 points**] Write pseudocode for an algorithm which, given an undirected graph G and a particular edge e in it, determines whether G has a cycle containing e. What is the runtime of this algorithm?

3. [**8 points**] Often there are multiple shortest paths between nodes of a graph. Write pseudocode for an algorithm that given an undirected, unweighted graph G and nodes $u, v \in V$, outputs the number of distinct shortest paths from u to v. What is the running time?

4. [**3 points**] Dijskstra's algorithm is optimal for finding the shortest distance from a starting node s to all other nodes in the graph *for graphs with positive edge weights.* Give an example graph with negative edge weights where Dijkstra's algorithm will give the wrong answer. Briefly explain your example.

5. [**5 points**] Someone suggests to you the following algorithm for finding the shortest distance (sum of edge weights) path from node s to node t in a directed graph with some negative edges: add a large constant to each edge weight so that all the weights become positive, then run Dijkstra's algorithm starting at node s, and return the shorted path found to node t.

   Is this a valid method? Argue that it works correctly or give a counterexample.