1 Introduction and Objective

Now that we have stored our graphs in one (or more – by now you should realize that you can do this) graph data structure(s) — or even not explicitly storing our graph (implicit graph), we want to run various (graph) algorithms on it.

In this tutorial, we will focus on two graph traversal algorithms: Depth-First Search (DFS) and Breadth-First Search (BFS) and concentrate on what they can do on top of just traversing the underlying graph.

We will heavily use https://visualgo.net/en/dfsbfs in this tutorial.

2 Tutorial 09 Questions

Prelude: BFS Review

Q0). As Prof Halim usually start with DFS first, we skipped BFS review during the previous Tut08. It is now time to properly review this important algorithm (will be revisited again for unweighted SSSP problem soon). Tutor can use https://visualgo.net/en/dfsbfs load any graph, and BFS from any randomly selected source vertex.

Review the Harder Topics

Q1). Tutor will spend some time (depending on the requests) to review any remaining harder topics about graph traversal that may not be clear even after Week 09+10 classes. In recent years, these are the usually harder topics for students, in decreasing order of difficulty:
Deeper Stuffs about Topological Sort

Q2). The modified DFS or modified BFS (Kahn’s) topological sort algorithm given in class (please review https://visualgo.net/en/dfsbfs ‘topological sort’, either the DFS or BFS version) only gives one valid topological ordering. How can we find all possible valid topological orderings for a given DAG? For example, there are 1008 possible valid topological orderings of the DAG in Figure 1. Starting point: What kind of DAG has the smallest/largest number of possible valid topological ordering, respectively?

Q3). The modified BFS (Kahn’s) topological sort algorithm is actually quite interesting (read the details at https://en.wikipedia.org/wiki/Topological_sorting#Kahn’s_algorithm). Can we change the underlying data structure (from a normal queue that is used in the modified BFS @ VisuAlgo) into another data structure? What if we replace the queue with a stack? What if we replace the queue with a priority queue? What if we replace the queue with a hash table?)

Graph Modeling Exercise Part 1

Q4). The tutor will randomly imagine one real life scenario (that can be modeled as a graph problem) and will ask random student to model that scenario into a graph. Students have to describe what the
set of vertices represent, what the set of edges represent, are the graphs weighted/directed/connected?, what are the graph (for now, limit to graph traversal) problem being asked?, should we explicitly store the graph?, etc...

**Hands-on 9**

TA will run the second half of this session with a few to do list:

- PS5 Debrief (A Quick One),
- Very quick review of Prof Halim’s
- Speedrun dfsbfs of VisuAlgo Online Quiz:
  [https://visualgo.net/training?diff=Medium&n=5&tl=5&module=dfsbfs](https://visualgo.net/training?diff=Medium&n=5&tl=5&module=dfsbfs)
- For CS2040S Only: Short PE Debrief
- Live solve another chosen Kattis problem involving Graph Traversal (toposort).

**Problem Set 6**

We will discuss some hints for the easier subtasks of the last PS6.