CS2040C Semester 1 2021/2022

Data Structures and Algorithms

Tutorial+Lab 08
Graph DS and Traversal

For Week 10

Document is last modified on: August 2, 2021

1 Introduction and Objective

This tutorial marks the last $\frac{1}{3}$ of CS2040/C/S: Graph. We will discuss various graph data structures and on how to explore them.

We will also review Union-Find Disjoint Sets (UFDS) data structure that has been discussed a few weeks back on Week 07.

The VisuAlgo pages that are used in this tutorial are `https://visualgo.net/en/graphds`, `https://visualgo.net/en/dfsbfs`, and `https://visualgo.net/en/ufds`.

2 Tutorial 08 Questions

Basic Stuffs About Graph DSes

Q1). (Not optional): Tutor will draw a *two-dimensional depiction* of **a random small** graph on the **whiteboard** first and ask students to store that graph in **either** Adjacency Matrix (AM)/Adjacency List (AL)/Edge List (EL) data structure on the **whiteboard**. Then, the tutorial group can compare the answers by re-drawing the same small graph on `https://visualgo.net/en/graphds`, possibly in different 2D depictions of the same graph to reinforce the concept that graph is a set of vertices and edges and can have many possible 2D depictions/embeddings.

Not-So-Basic Stuffs About Graph DSes

Q2a). Draw a Directed Acyclic Graph (DAG) with V vertices and $V \times (V-1)/2$ directed edges.

Q2b). Draw a Bipartite Graph with V vertices (assume that V is even) and $V^2/4$ undirected edges.

Q2c). Draw a Tree with V vertices (and $E = V - 1$ edges) that is not a Bipartite graph.

Q3). Show what is the best (fastest) way to convert a graph currently stored in graph data structure 1 into graph data structure 2.

Q3a). From Adjacency Matrix (AM) to Adjacency List (AL)

Q3b). From AM to Edge List (EL)

Q3c). From AL to AM

Q3d). From AL to EL

Q3e). From EL to AM

Q3f). From EL to AL

For the interest of time, tutor will only pick subset of these for live discussion (the rest are documented in modal answers)

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?, etc...

DFS/BFS Review

Q5). (Not optional): The tutor will end the tutorial by quickly reviewing DFS and BFS graph traversal algorithms using `https://visualgo.net/en/dfsbfs` starting from the same randomly drawn graph post Q1). discussion. The tutor will ask some students to join the live demonstration. We will discuss harder applications of these two graph traversal algorithms in Tut09.

UFDS Review

Q6). (Optional if needed). Quickly review the findSet, isSameSet, unionSet operations in `https://visualgo.net/en/ufds`.

Q7). Back in Week 07, we discussed a one-off data structure with specific application: Union Find Disjoint Sets. It has not been properly discussed in Tut06 and Tut07. One of the most commonly used application of this UFDS data structure is to find Connected Components (CCs) of an undirected unweighted graph. Contrast this UFDS data structure with DFS/BFS approach for the same application (finding CCs).

Hands-on 8

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

- Very quick review of Steven's
  `https://github.com/stevenhalim/cpbook-code/blob/master/ch2/ourown/graph_ds.cpp`,

- Very quick review of Steven's
  `https://github.com/stevenhalim/cpbook-code/blob/master/ch4/traversal/dfs_cc.cpp`
  and `https://github.com/stevenhalim/cpbook-code/blob/master/ch4/sssp/bfs.cpp`,

- Do a(nother) sample speed run of VisuAlgo online quiz that are applicable so far, e.g.,
  `https://visualgo.net/training?diff=Medium&n=5&tl=0&module=graphds,dfsbfs`.

- Then, live solve another chosen Kattis problem involving Graph Data Structure.

Problem Set 4 (Again)

We will end the tutorial with **high-level** discussion of PS4.
Restrictions are now lifted.
TAs can discuss the **high-level** ideas to get full marks for PS4 A+B, hence it is just an 'implementation issue' from here onwards.

WARNING: Plagiarism checker is strongly enforced this time and if anyone is still copy/seek too much help for the much longer (≈ 2 weeks) PS, you can say goodbye to lots of marks for the 2 hours PE on Week 11.