CS2040C Semester 2 2021/2022

Data Structures and Algorithms

Tutorial+Lab 08
Augmented Trees, Start of Graph DS and Traversal, UFDS Review 2

For Week 10

Document is last modified on: March 14, 2022

1   Introduction and Objective

In this tutorial, we will first discuss bBST augmentations by discussing operations that utilizes its 'ordered' property: Select and Rank, as well as a simple augmentation to make the standard bBST able to handle duplicate elements.

Then we will transition to the last $\frac{1}{3}$ of CS2040/C/S: Graph. We will discuss various graph data structures and on how to explore them with basic graph traversal algorithms.

The VisuAlgo pages that are used in this tutorial are `https://visualgo.net/en/avl` (again), `https://visualgo.net/en/graphds`, `https://visualgo.net/en/dfsbfs`, and `https://visualgo.net/en/ufds` (to contrast with dfsbfs for one application).

2   Tutorial 08 Questions

Extra BST Operations (After Augmentations)

Q1). There are two important bBST operations: Select and Rank that are not included in VisuAlgo yet (overview at `https://visualgo.net/en/bst?slide=5-1`) but can be quite useful for some **order-statistics** problems. Please discuss on how to implement these two operations efficiently after we augment each bBST vertex with an 'extra attribute'.

Q2). What if there are duplicate elements in our bBST? Please discuss on how to implement this feature efficiently after we augment each bBST vertex with yet another 'extra attribute' (different from above). This is particularly important for std::multiset or std::multimap implementations.

Basic Stuffs About Graph DSes

Q3). (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

Q4). (Choose 2 out of 3) Many of these are in VisuAlgo online quiz, get ready:
Q4a). Draw a Directed Acyclic Graph (DAG) with V vertices and $V \times (V-1)/2$ directed edges.
Q4b). Draw a Bipartite Graph with V vertices (assume that V is even) and $V^2/4$ undirected edges.
Q4c). Draw a Tree with V vertices (and $E = V - 1$ edges) that is not a Bipartite graph.

Q5). (Choose 2 out of 6) Show what is the best (fastest) way to convert a graph currently stored in graph data structure A into graph data structure B.
Q5a). From Adjacency Matrix (AM) to Adjacency List (AL)
Q5b). From AM to Edge List (EL)
Q5c). From AL to AM
Q5d). From AL to EL
Q5e). From EL to AM
Q5f). From EL to AL
For the interest of time, tutor will only pick subset of two of these for live discussion (the rest are documented in modal answers)

DFS/BFS Initial Review

Q6). (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 from Q3). 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

Q7). Back in Week 07, we discussed a one-off data structure with specific application: Union Find Disjoint Sets. Now we want to connect this data structure with one specific graph application: 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 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 BST/AVL Tree/augmented BST...

Problem Set 5 (Again)

We will end the tutorial with **high-level** discussion of PS5.

Restrictions are now lifted.

TAs can discuss the **high-level** ideas to get full marks for PS5 A+B, hence it is just an 'implementation issue' from here onwards.

WARNING: Anyone who still relying too much on TA/peer hints to get unstuck in the much longer (≈ 2 weeks) PS need to prepare to say goodbye to lots of marks for the 2 hours PE on Week 11.