1 Introduction and Objective

Welcome back from recess week =). I hope you are a bit fresher now (regardless of your Written Quiz 1 scores).

In this tutorial, we will first do a debrief of the Written Quiz 1 (WQ1) that was conducted on Saturday, 19 September 2015. Your tutor will discuss the typical common mistakes, make the WQ1 questions even harder :O, and share with you how to answer certain questions in a better (shorter) ways. Hopefully everybody will get his/her own closures after this ‘very eye-opening’ WQ1.

We will then resume our discussion on Graph problems by reviewing Breadth-First Search (BFS) and Depth-First Search (DFS) graph traversal algorithms using [http://visualgo.net/dfsbfs.html](http://visualgo.net/dfsbfs.html), followed by just one very interesting past final exam paper (another Steven’s best question, comparable with the Section 3.2. of WQ1). Finally, the tutor will discuss the trivial idea to solve PS3 Subtask A before dismissing the class.
2 Tutorial 05 Questions

Written Quiz 1 Debrief

The tutor will spend 20 minutes to:

1. Return your WQ1 script, if you have not collected it earlier.

2. Ask you to check if everything is in order and your score has been entered correctly at [http://www.comp.nus.edu.sg/~stevenha/cs2010.html#leaderboard](http://www.comp.nus.edu.sg/~stevenha/cs2010.html#leaderboard) (you need to login using your CS2010 account as this data is not displayed publicly).

3. Discuss common mistakes and the general cases of the question (if any), e.g.

   (a) 10000 Booleans? Can you use get $O(1)$ bit manipulation performance?
       No, it is no longer $O(1)$, but up to 10000 steps. Even if you say that you will use 10000/64 = 157 64-bit integers, it is still 157 steps, no longer ‘instantaneous’.

   (b) Can a graph change status from directed to undirected dynamically?
       Yes, start with an undirected graph (all edges are bidirectional), add one directed edge, it is now a directed graph, delete it again, it is now an undirected graph again. It is similar to PS3 where we start from a connected graph and once we delete a vertex, we (may) disconnect the graph, so the status of the graph can change due to addition/deletion of a vertex/edge.

   (c) General AVL Tree insertion balancing act challenge (what if $n = 15$? $n = 31$?)
       With a systematic strategy, this question is easy.

   (d) AVL Tree deletion domino effect challenge (what if Steven asks for 4 (FOUR) group of rotations, or even 5 (FIVE) groups... can you find the required pattern now?),
       With a systematic strategy, this question is also easy.

   (e) UFDS tree challenge (short tree, tall tree, tree of certain specified height?),
       With a systematic strategy, this question is also easy.

   (f) Leaderboard system (what’s the purpose of the 4th AVL or an extra HashMap?, Can BinaryHeap be used to answer GetMedian effectively? What if Steven asks for GetPercentile([0..100])? Is median always the root of an AVL tree?
       This question simply tests your combined understanding of PS1+PS2.

   (g) And of course, the eye opening last question :).
       This question is for the top end of the class.

BFS/DFS Review

Then the tutor will spend not more than 5 minutes to quickly review BFS/DFS graph traversal algorithm using [http://visualgo.net/dfsdfs.html](http://visualgo.net/dfsdfs.html) starting from a randomly drawn graph.

This part is left to the tutor.
Another Out of the Box Thinking Question

Please download CS2010 Final Exam paper S1, AY2013/2014 and solve a problem titled: Facebook Privacy Setting (19 marks)

Simply run the merge subroutine of mergesort algorithm between the sorted AdjList[i] and AdjList[j]. As soon as the frontmost items of the two sorted lists are the same, we have a common friend between i and j (degree 2). This is $O(k)$. Checking degree 1 or 0 are trivial :).

Problem Set 3

Finally, the tutor will quickly discuss the trivial idea for PS3 Subtask A, especially for those who have not cleared this 50 points Subtask A.

In Subtask A, you are given a tree... In a tree, any deletion of internal vertex will disconnect it, so simply check the given graph data structure and find vertices of degree 2 (or more). Among those vertices, pick the one with the least rating score... Done :O.