CS2040S Recitation 5

AY 24/25 Sem 2 — github/omgeta

Q1. (a.) Create nodes where the key is height (double), and values are name (string) and grade (double)

(b.) Use an AVL tree to maintain a balanced BST. Augment with the sum of grades in each subtree.

(c.) Modify each node with a list of student information.

Q2. (a.) Use an AVL tree where the nodes are inserted with the initial flip state and the key is their index. Augment each node with a lazy flag indicated that the nodes in the subtree should be flipped. Now, *turnOver* just finds the subtree root node in $O(\log n)$ and toggles the lazy flip flag. When the actual flip state of the card is evaluated, just trace the number of flips along the path to the node.