CIS 313, Intermediate Data Structures
Fall 2020

Assignment 4

Due: December 4, 2020

1. 18.1-3 (p.491) [**5 points**]

2. 18.2-3 (p.497) [**5 points**]

3. Insert into an initially empty 2-3-4 tree (aka, a B-tree with minimum degree 2), in the order given, the following values:

   $$12,\ 13,\ 17,\ 10,\ 4,\ 6,\ 9,\ 15,\ 30,\ 25,\ 20,\ 40.$$

   Show the intermediate trees after each insertion that causes a split. [**10 points**]

4. Suppose you have an array S of size n, where each element in S represents a different vote for class president, where each vote is given as an integer representing the student ID of the candidate. Without making any assumptions about who is running or how many candidates there are, design an $O(n \lg n)$ algorithm to determine which candidate receives the most votes. [**10 points**]

5. Consider a modification to the previous problem to a situation where we know the number $k < n$ of candidates running. Design an $O(n \lg k)$ algorithm to determine which candidate receives the most votes. [**10 points**]

6. Given the input of the first problem, give an $O(n)$ time algorithm to determine if some candidate received a majority ($\lceil \frac{n+1}{2} \rceil$) of the votes.

   - $O(n)$ average or expected time is OK
   - so the hint is to look for the median

   [**10 points**]

Total: 50 points