

CS304 Assignment 4: Trees and Heaps - Due Wednesday March 22nd at 11:59 PM

Part 1: *Written questions: do these questions using pencil and paper. Take photograph and upload in pdf format to moodle along with your c++ code files from part 2.*

1a: inserting into BST:

Build a BST by inserting the following sequence into an empty BST: [5,2,1,6,3,7,4,8]. Show only the final BST.

1b removing from BST:

Show the BST from 1a after each of the following operations: `remove(2)`, `remove(5)`, `remove(3)`. (Show the tree after each operation. The final tree will have only five elements remaining).

1c: inserting into AVL tree:

Build an AVL tree by inserting the following sequence into an empty AVL tree: [3,2,1,4,5,6,7,26,25,24,20,18,12]

1d: removing from AVL tree:

Show the AVL tree from 1c after each of the following operations: `remove(24)`, `remove(1)`, `remove(12)`, `remove(5)`

1e: inserting into heap:

Build a min heap by inserting the following elements into an empty heap: [4,3,2,1,9,8,7,6,5,0]

1f: removing from heap:

Show the heap from 1e after 8 separate calls to `deleteMin`. Show the heap's state after each call to `deleteMin`.

Part 2: *coding question and timing experiment. You may use code from the textbook.*

Heapsort vs Treesort

In part two, you will compare the performance of two new sorting algorithms (heapsort and treesort). Treesort is an algorithm that works by first creating a BST from the unsorted input sequence, and then outputting the sorted result by applying an inorder traversal to the BST. Heapsort works by first building a heap from the unsorted input sequence, and then repeatedly calling `deleteMin` on the heap until the heap is empty, outputting the result of each `deleteMin` into the sorted array.

Part 2a: write C++ code to perform heapsort and treesort on an array of $n = 100000$ randomly generated numbers. You can use the code from the textbook for the BST and heap, but you will need to add more code to complete the sorting experiments (for example, you will need to add the inorder traversal method to the BST).

Part 2b: repeat 2a, but use an AVL tree instead of a BST for treesort.

Part 2c: discuss the results of heapsort and treesorts vs other sorting algorithms (quicksort, mergesort).