

Niema Widaha

## MAC 286: Data Discussion 1: Part 2

1. What do the two responses agree on? Disagree on?
  - a. Both responses agree on the fact that the process of finding a particular exit can be an expensive and an inefficient process if the best solution is not optimized.
  - b. They disagree on which structure is the best one to utilize. The first prompt believes that the best solution would be to implement a binary search tree, due to its efficient insertion and being able to craft the binary search tree with methods that allow for true optimization. The first response also notes that the solution in the second response of utilizing arrays is a less efficient approach.
2. Is there anything you agree with in your peer's comments? Disagree with?
  - a. I agree with the facts listed in the first prompt, that the time complexity for the insert and search operations is  $O(\log(n))$ . I also agreed with the measurement algorithm they've blueprinted to find the distance from the border. I like that they've taken the time to identify the approaches needed to implement this prompt with the binary search tree, and what exactly this would cost the program in terms of speed and access to information.
  - b. One approach that I disagree with is that in the second response they'd like to utilize both ordered arrays and a binary search, which is not correctly answering the prompt because they're using two data structures to implement the solution.
3. What would your chose of a data structure and implementation in this situation be now and is it any different from before? Why/why not?
  - a. My choice of a data structure would also be a binary search tree, because I agree with the first response but also because it's the most efficient approach. I do believe that the linked list implementation can also be another solution but a less efficient one due to its time complexity. With the linked list, I was able to store the obtained information needed for each exit in the Node class by building a multi-faceted constructor to hold the String exit\_name and int exit\_number. I also would implement the methods from my linked list with a binary search tree, like the findTheNearestExit(ExitNode currentExit) and the distanceFromBasepoint(ExitNode currentNode) method.