

# Assignment #4: BST's

9/27/2023

## 9.5/10 Points

Attempt 1



Review Feedback  
9/27/2023

Attempt 1 Score:  
9.5/10



View Feedback

Anonymous Grading: no

### Unlimited Attempts Allowed

#### Details

Build a Binary Search Tree, as balanced as possible; you should think about how to do this, without using AVL properties/rotations, from the integers (in no specific order):

[6, 17, 20, 41, 45, 52, 57, 65, 71, 76, 79, 87, 92, 95, 99]

Find the k-th smallest element in the BST (you should think about how to do this).

Input: (the given set of integers in an array ; user-defined k)

Output: print the BST (you can choose the format) and return the k-th smallest element

Submit:

a single file, **X\_Y\_PA4.zip** or **X\_Y\_PA4.tar.gz**, where X is your first name (capital letters) and Y is your last name (capital letters). You get this file from compressing a folder named **X\_Y\_PA4** containing the following:

- Your code files (.cpp, .h, etc.)
- A Makefile file that contains all the commands needed to compile your code on **Tesla**. All the code will be tested on Tesla with its g++. I should be able to compile your code by executing **make**
- A **README.txt** file showing how the users should execute/run your program
- A screen recording of your compilation/execution

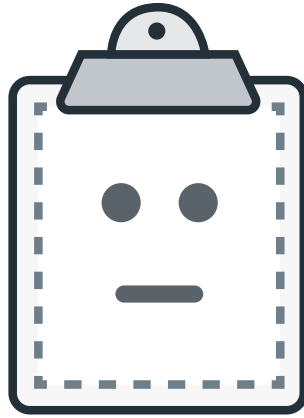
Point reduction:

-2pts for no automated initialization of balance

-3pts for no k-th smallest element calculation

-1pts for k not being user defined (input)

-1pts for not submitting a screen recording



Preview Unavailable

PARMINDAR\_SINGH\_PA4.zip.zip



[Download](#)

([https://iu.instructure.com/files/162868274/download?download\\_frd=1&verifier=6C8x3ihhWyzKZwk7rnIW3ZbzUBtwt6JKggeGAu8j](https://iu.instructure.com/files/162868274/download?download_frd=1&verifier=6C8x3ihhWyzKZwk7rnIW3ZbzUBtwt6JKggeGAu8j))

*You are unable to submit to this assignment as your enrollment in this course has been concluded.*