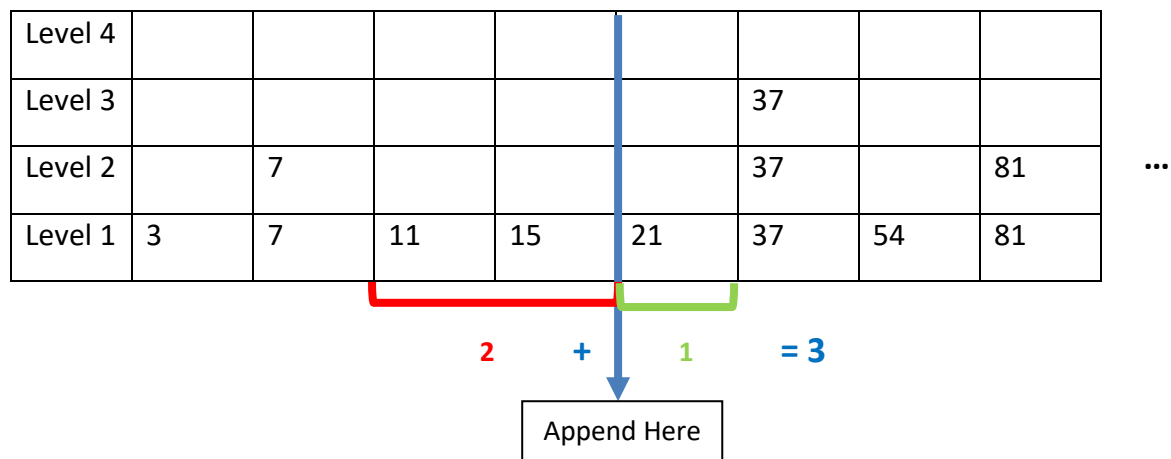# GTU Department of Computer Engineering
CSE 222/505 - Spring 2022
Homework 7
Due date: May 12, 2022– 23:55

Q1. Write a method that takes a binary tree and an array of items as input, and it returns a binary search tree (BST) as output. The binary tree contains *n* nodes and doesn't need to be balanced. The array contains *n* unique items which are mutually comparable. The method should build a binary search tree of *n* nodes. The binary search tree should contain the items. The structure of the binary search tree should be same as the structure of the binary tree. Explain your function in detail and give theoretical run time complexity analysis of that method in the report.

Q2. You are asked to write a method that takes a binary search tree (BST) as a parameter and returns the AVL tree obtained by rearranging the BST. The method should convert the BST into an AVL tree by using rotation operations. Explain your function in detail and give theoretical run time complexity analysis of that method in the report.

Q3. In this question, you are expected to develop a novel version of the skip list data structure, CustomSkipList. There are *k* levels in this skip list, and *k* increases by the number of items in the first-level list. A new level list is added to the skip list each time when the size of the first level list reaches powers of 10. The tall items (contained in the more than one level) are appended to a one-level upper list when a new level is added to the skip list. The CustomSkipList has two level lists in default, and it is customizable. An item can be also appended to upper-level lists during insertion operation. The item is always appended to the first level list, and it is appended to higher levels by increasing chance with the number of items in the range between its closest tall neighbors on the left side and the right side. The probability of appending an item to a higher level is calculated by dividing the number of items between two tall neighbors by 10. Thus, if the number of items between two tall neighbors of an item is 5, then the chance of adding this item to an upper-level list is 50%. See the example below for better understanding. The chance is constant for all the upper levels. All other operations like the searching strategy are the same as in the regular skip list data structure explained in the textbook. Analyze the run time complexity of both adding a new item and adding a new list operations, and explain with details in the report file.

Example: Add 19 to the skip list.

Level 4								
Level 3						37		
Level 2		7				37		81
Level 1	3	7	11	15	21	37	54	81

2 **+** **1** **= 3**

Append Here

The chance of adding the item to the higher levels is 3 / 10 * 100 = 30%.

*P.S.: Consider only one side if the item will be appended to the head or tail of the list.

GENERAL RULES:

- For any question firstly use the forum on the MS Teams page, and then the contact TA.
- You can submit an assignment one day late and will be evaluated by over sixty percent (%60).

TECHNICAL RULES:

- You must write a driver function that demonstrates all possible actions in your homework. For example, if you are asked to implement an array list and perform an iterative search on the list then, you must at least provide the following in the driver function:
    o Create an array list and add items to the list. Append items to the head, tail, and k^{th} index of the list.
    o Perform at least two different searches by using two items in the list and print the index of the items.
    o Perform another search with an item that isn't in the array list and inform the user that the item doesn't exist in the array list.
    o Delete an existing item from the list and repeat the searches.
    o Try to delete an item that is not on the array list and throw an exception for this situation.

  The driver function should run when the code file is executed.

- Implement clean code standards in your code;
    o Classes, methods, and variables names must be meaningful and related to the functionality.
    o Your functions and classes must be simple, general, reusable, and focus on one topic.

o   Use standard java code name conventions.

REPORT RULES:

- Add all javadoc documentations for classes, methods, variables …etc. All explanations must be meaningful and understandable.
- You should submit your homework code, Javadoc, and report to MS Teams in a "studentid_hw1.tar.gz" file.
- Use the given homework format including selected parts from the table below:

Detailed system requirements	✓
The Project use case diagrams	X
Class diagrams	✓
Other diagrams (extra points)	X
Problem solutions approach	✓
Test cases	✓
Running command and results	✓

GRADING :

- **No OOP design:**   **-100**
- **No error handling:**   **-50**
- No javadoc documentation: -50
- No report: -90
- Disobey restrictions: -100
- **Cheating:**   **-200**
- Your solution is evaluated over 100 as your performance.

CONTACT :

- Teaching Assistant Ferda Abbasoğlu, ferdaabbasoglu@gtu.edu.tr