You are expected to work on the problems before coming to the lab. Discussion sessions are not meant to be a lecture. TA will guide the discussion and correct your solutions if needed. We will not release "official" solutions. If you are better prepared for discussion, you will learn more. TAs will record names of the students who actively engage in discussion and report them to the instructor. The instructor factors-in participation in the final grade as bonus points.

1. Exercise 10.1-1.
2. Exercise 10.1-2.
3. Exercise 10.1-3.
4. Exercise 10.1-4.
5. Exercise 10.1-5.
6. Exercise 10.1-6.
7. Exercise 10.1-7.
8. Exercise 10.2-2.
9. Exercise 10.2-3.
10. Exercise 10.2-4.
11. Exercise 10.2-6.
12. Exercise 10.2-7.
13. Exercise 12.1-1.
14. Exercise 12.1-2.
15. Exercise 12.1-3.
16. Exercise 12.1-4.
17. Consider the binary search tree (BST) in Fig 12.1.(b). Print all the keys in the BST in preorder, then postorder.
18. Exercise 12.2-1.
19. Exercise 12.2-2.
20. Exercise 12.2-3.
21. Exercise 12.2-5.
22. Exercise 12.2-6.

23. Exercise 12.3-3.

24. Exercise 12.3-4.

25. Exercise 12.3-6.

26. We want to add a new operation to BST, namely AD (Ancestor-Descendant), which takes as input two nodes x and y and outputs true if x is y's ancestor, otherwise false. You can assume that $x \neq NIL$ and $y \neq NIL$. Write the pseudocode for AD.

27. We want to add a new operation to BST, namely Sum, which takes as input a node x (or more precisely the pointer to the node), and outputs the sum of the key values of all nodes in the tree rooted at x. Your algorithm must run in linear time.

28. You are given a *sorted* array $A[1 \ldots n]$. Give a linear time algorithm that constructs a BST of height $O(\log n)$ with $A[1 \ldots n]$ as key values. Give the pseudo-code.

29. Problem 12-4.