German University in Cairo
Media Engineering and Technology
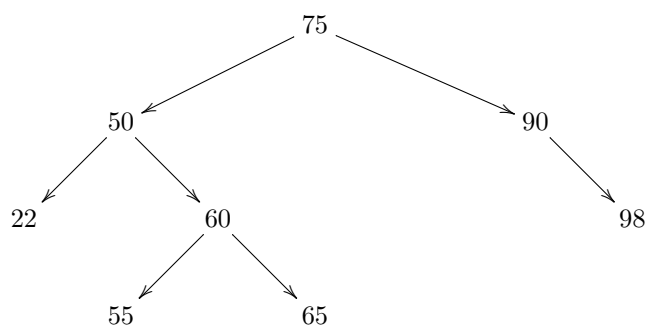Prof. Dr. Slim Abdennadher

Data Structures and Algorithms, Winter term 2020
Practice Assignment 10

Exercise 10-1   Traversal

a) Given the following binary search tree:

   1. Give the preorder, postorder and inorder traversals of the tree above.
   2. Which traversal order lists the info fields of the tree nodes in sorted fashion?

b) Given the following output of the preorder traversal of a binary search tree:

   `90, 50, 20, 5, 25, 75, 66, 80, 150, 95, 92, 111, 175, 166, 200`

   Construct the binary search tree. Is the tree **unique**? Show your workout.

c) Given a preorder traversal and an inorder traversal of a binary tree, a unique binary tree that satisfies the traversals can be constructed. Given the following preorder and inorder traversals, construct the binary tree:

   - **Preorder Traversal**: `a, e, f, h, g, b, c, d`
   - **Inorder Traversal**: `h, f, e, g, a, c, b, d`

Exercise 10-2   More Traversals

It is possible to construct the unique tree that **each pair** of traversals represent?

a) Inorder traversal: `2 4 9 5 8 3 1`
   Postoreder traversal: `4 5 9 2 1 3 8`

b) Inorder traversal: `9 8 6 1 2 5 4`
   Postorder traversal: `9 6 1 8 5 4 2`

c) Inorder traversal: `4 5 6 7 3 2`
   Preorder traversal: `2 3 4 7 6 5`

d) Preorder traversal: `1 2 3`
   Postorder traversal: `3 2 1`

Exercise 10-3   Logarithmic Search in a BST (Final Exam 2013)

Suppose that we have numbers between 1 and 100 in a binary search tree and want to search for the number 45 using a logarithmic algorithm. Which (possibly multiple) of the following sequences could be the sequence of nodes examined before reaching 45? Justify your answers for every sequence.

a) `5, 2, 1, 10, 39, 34, 77, 63`

b) `1, 2, 3, 4, 5, 6, 7, 8`

c) `9, 8, 63, 0, 4, 3, 2, 1`

d) `8, 7, 6, 5, 4, 3, 2, 1`

e) `50, 25, 26, 27, 40, 44, 42`

f) `50, 25, 26, 27, 40, 44`

Exercise 10-4   Level Printing

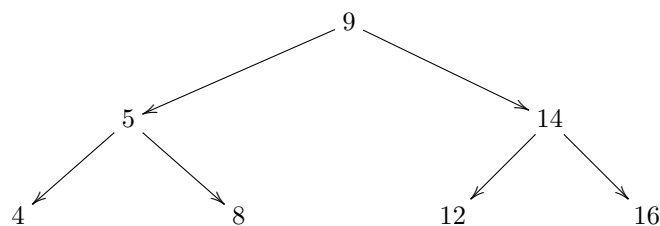Implement a method that prints all the nodes in a certain level For example:
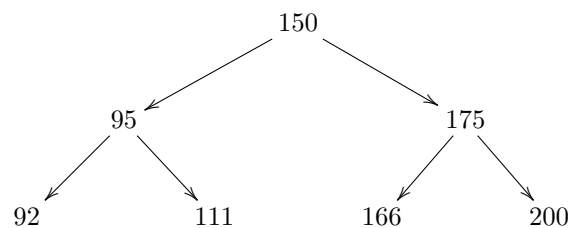
invoking `printLevel(2)` on the binary tree above should print: `92 111 166 200`
Hint: you can pass another parameter to the method.

Exercise 10-5   To be solved in the Lab

Write the method `breadthTraversal` that will print all the nodes in the tree level by level.
For example:

invoking `breadthTraversal()` on the binary tree above should print: `9 5 14 4 8 12 16`

Exercise 10-6   Descending Order

a) Given a binary search tree, how can you perform a traversal which will produce the values ordered in **descending order** (largest to smallest)? Show your workout on traversing the tree below.

4

2   5

1   3   11

10
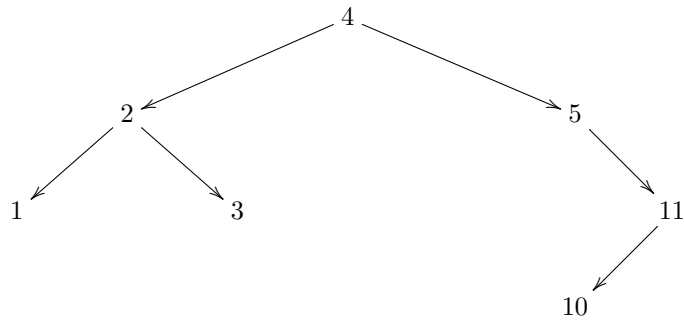
b) Write a **recursive** method `String descendingOrder()` that performs this traversal.

Exercise 10-7   Range Printing

Write a method `void printRange(int lowerBound, int upperBound)` that prints all keys k in a binary search tree where $lowerBound \leq k \leq upperBound$. For example:

50

20   75

5   25   66   80
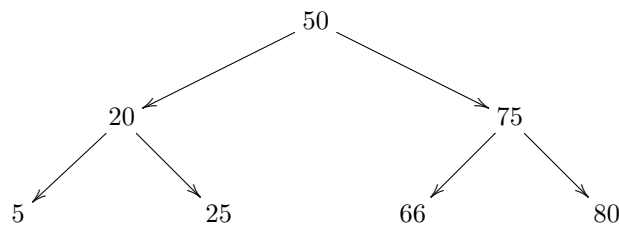
invoking `printRange(20, 90)` on the binary tree above should print: `50 20 25 75 66 80`

Exercise 10-8   Preorder Traversal (Iterative)

Write an **iterative** method `void preorderIter()` in class `BinaryTree` that prints the preorder traversal.

Hint: Consider using stacks.

Exercise 10-9   Path in a Tree

Write a method `LinkList path(Comparable key)` that returns a linked list starting from the root, including all nodes on the path to a certain target node (with a value `key`) and including the target node.

Exercise 10-10   Height

Levels in a tree are numbered starting with 0 for the root node level and increasing downwards. The number of the highest level is called the *height* or *depth* of the tree.

   a) Write a **recursive** method to calculate the height of a binary search tree.

   b) Write an **iterative** method to calculate the height of a binary search tree.

   c) What is the **smallest** number of nodes a binary tree of height n can have? Show your workout.

   d) What is the **largest** number of nodes a binary tree of height n can have? Show your workout.

Exercise 10-11   Ternary Tree

A ternary tree is a tree where each node can have **at most 3** children.
Consider a ternary tree of height 3:

   a) What is the minimum number of nodes?

   b) What is the maximum number of nodes?