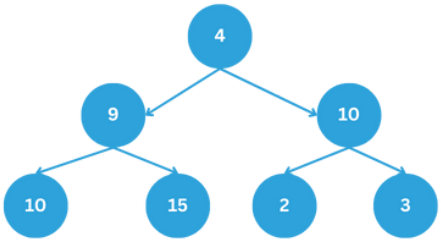
German International University
Computer Science and Informatics
Dr. Nada Sharaf

Computer Organization, Spring term 2024
Project Announcement

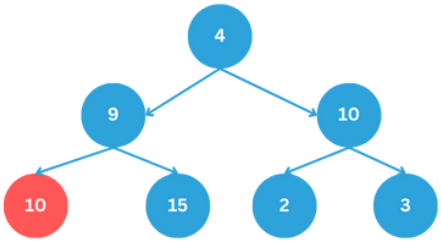
Deadline: 23/05/2024

A tree is a data structure to save data in nodes. Every tree has a root (the top element). In a binary tree, every node can have two children maximum as shown below

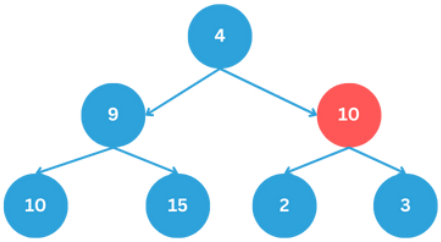


You are requested to implement two tree-search algorithms. The first is a depth first search. In this case, you should return the level at which the element was found. If the element is not found, you should return -1.

For example, if you are searching for 10, the depth-first returns 3 because it is found at level 3 as highlighted below:

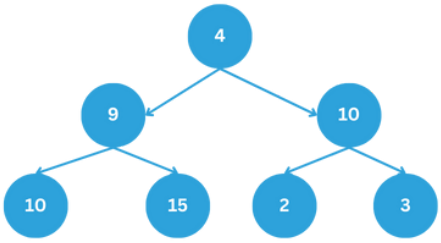


However, if you are searching for 10, with the breadth-first returns 2 because it is found at level 2 in this case as highlighted below:



Saving a Tree

We assume that the elements of the tree are going to be saved in an array.



There are two ways to save the elements:

a) Representation 1: The tree shown above could be saved in the following array `[4,9,10,10,15,2,3]`.

b) Representation 2: Another way is to use this array: `[4,9,10,15,10,2,3]`

Requirements

You are required to implement depth-first and breadth-first search using the two possible representations. You are also required to implement a procedure that converts from one representation to the other

a) Procedure 1: convert from Representation 1 to Representation 2. The output is an array with the new representation. The address of the new array is returned.

b) Procedure 2: convert from Representation 2 to Representation 1. The output is an array with the new representation. The address of the new array is returned.

c) Procedure 3: Take a value as an input in addition to the tree with Representation 1 and do breadth-first search.

d) Procedure 4: Take a value as an input in addition to the tree with Representation 2 and do breadth-first search.