

Problem: K-tree is a tree that each node has at most k children. Binary tree is a special case of k-tree where $k = 2$. Design an algorithm to convert k-tree to 2-tree that can convert back to original k-tree.

Variables:

Node k-root

Node b-root

int branch // branch of b-tree, 0 = left or 1 = right

1. Converting k-tree to 2-tree

Time Complexity: $O(n)$

Space Complexity: $O(n)$

Default direction: left

Original direction: branch

Opposite direction: 1^{branch}

In each recursion:

1. Iterate through each child of k-root.children
2. Moves to the next child in b-root (original direction)
3. Replace value of b-root by value of the current child
4. As we move to the next depth, the direction of 2-tree is changed

PSEU-DO CODE

TO-2-TREE(k-root, b-root, branch)

if k-root == null
return null

temp = b-root

for each child in k-root.children:

b-root = b-root.child at branch

b-root.value = child.value

TO-2-TREE(child, b-root, 1^{branch}) // switch to the other branch

return temp // return the original b-root

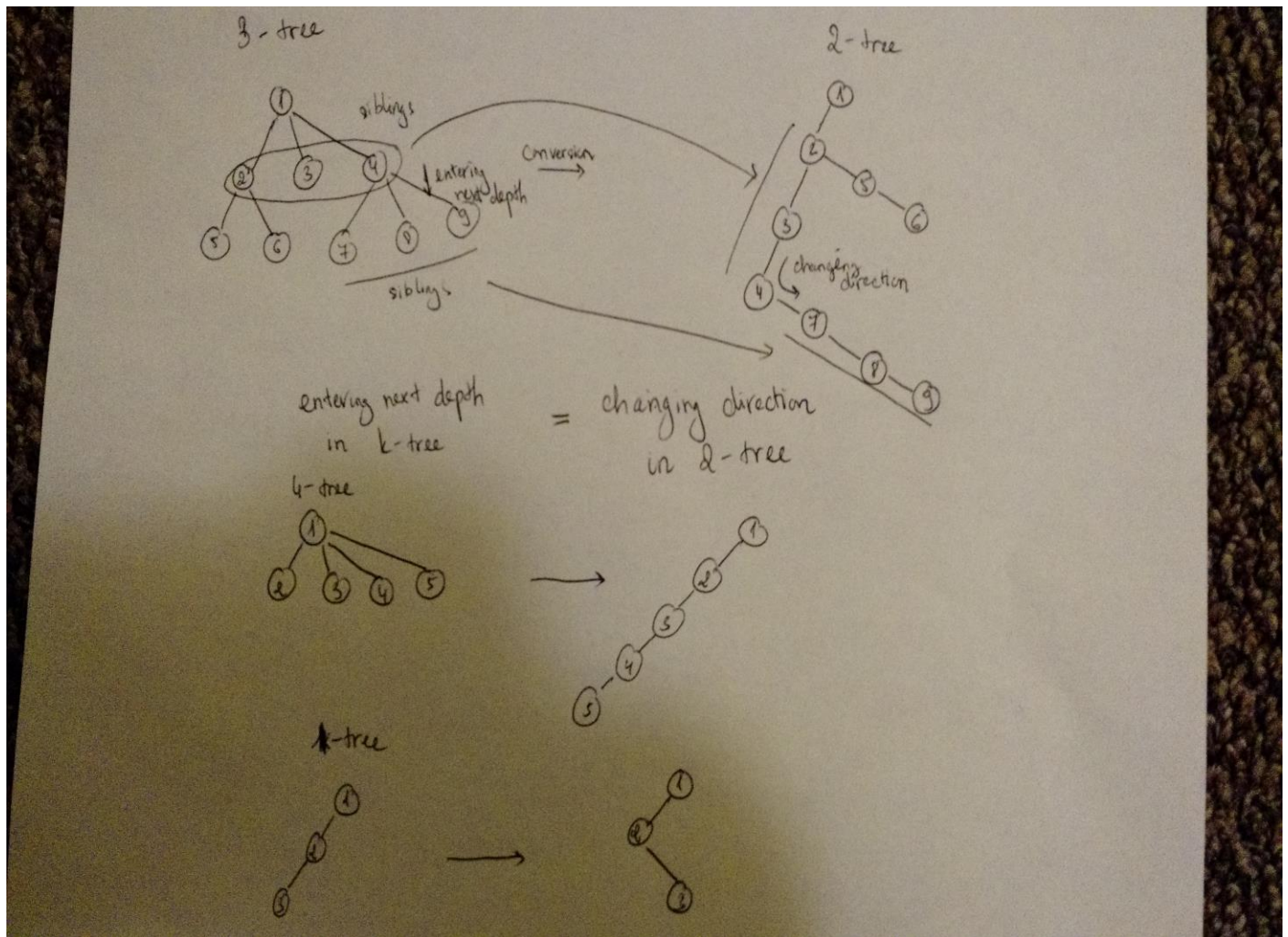


Figure 1

Figure 1 shows three examples of how k-tree is converted to 2-tree. The direction changes whenever we enter the next depth.

2. Retrieving k-tree from 2-tree

Time Complexity: $O(n)$

Space Complexity: $O(n)$

Default direction: left (as constructed from to2Tree function)

Variable branch represents the direction of all siblings in k-tree

Two cases of direction: left-right or right-left

In each recursion:

1. Add b-root.value to k-root.children
2. Recur the newly added child with the opposite direction (must be done before step 3)
3. Recur with the original direction to add new value from 2-tree to k-root.children

PSEU-DO CODE

TO-K-TREE(b-root, branch, k-root)

if b-root == null
return null

add b-root to k-root.children

// recur in the following order

TO-K-TREE(b-root.child at $1^{\wedge}branch$, $1^{\wedge}branch$, newly added child in k-root)

TO-K-TREE(b-root.child at branch, branch, k-root)

return k-root

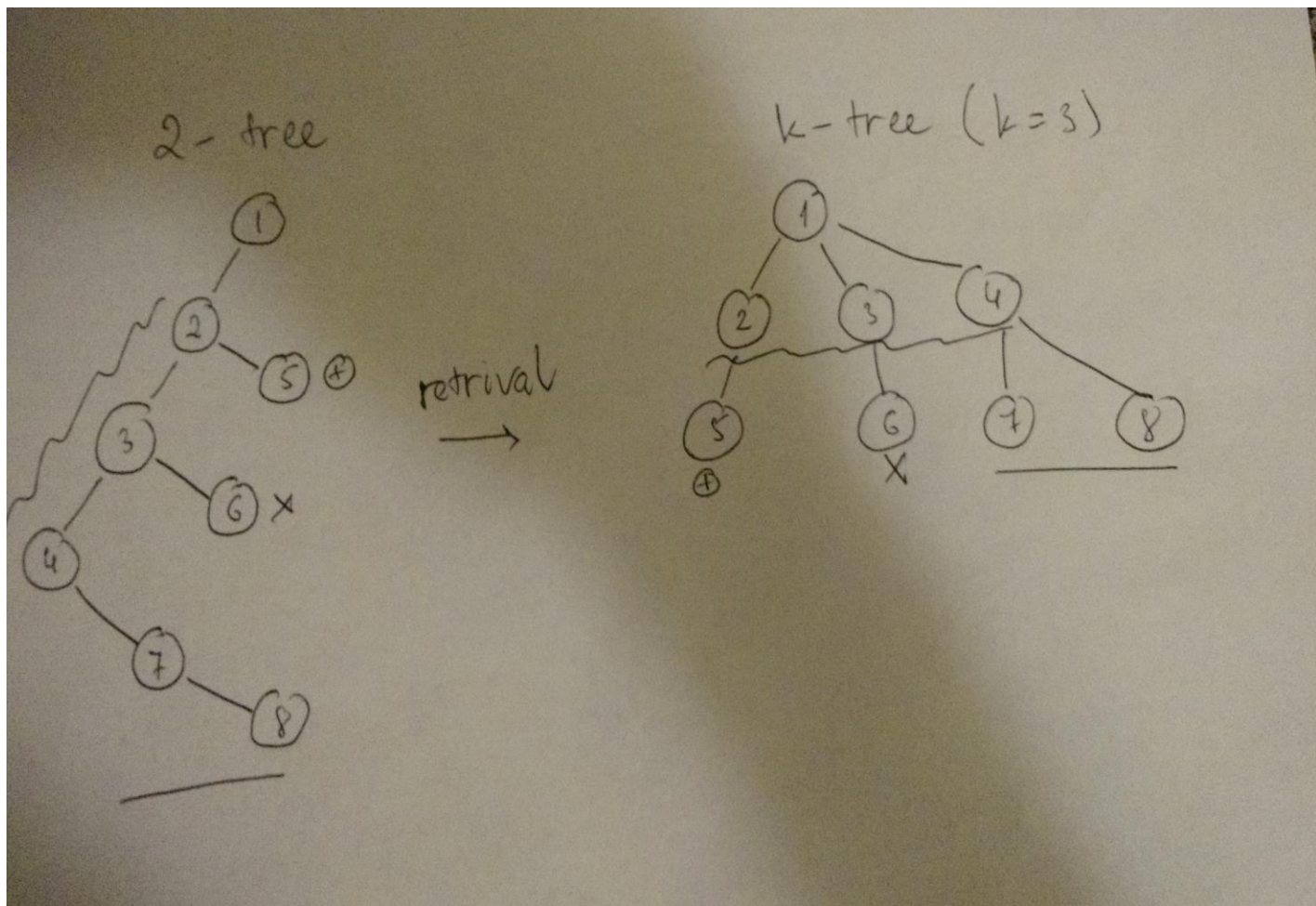


Figure 2

Figure 2 shows an illustration of the retrieval algorithm, each mark in k-tree is mapped with an identical mark in 2-tree to show relationship between siblings and depths.