B-Tree

[attribute]

|  |  |  |
| --- | --- | --- |
| Tree |  | Height of tree |
| Every node |  | The numbers of keys stored in node x. |
|  | The keys |
|  | A Boolean value check the node x is leaf node or not? |
| Internal node |  | Its children |

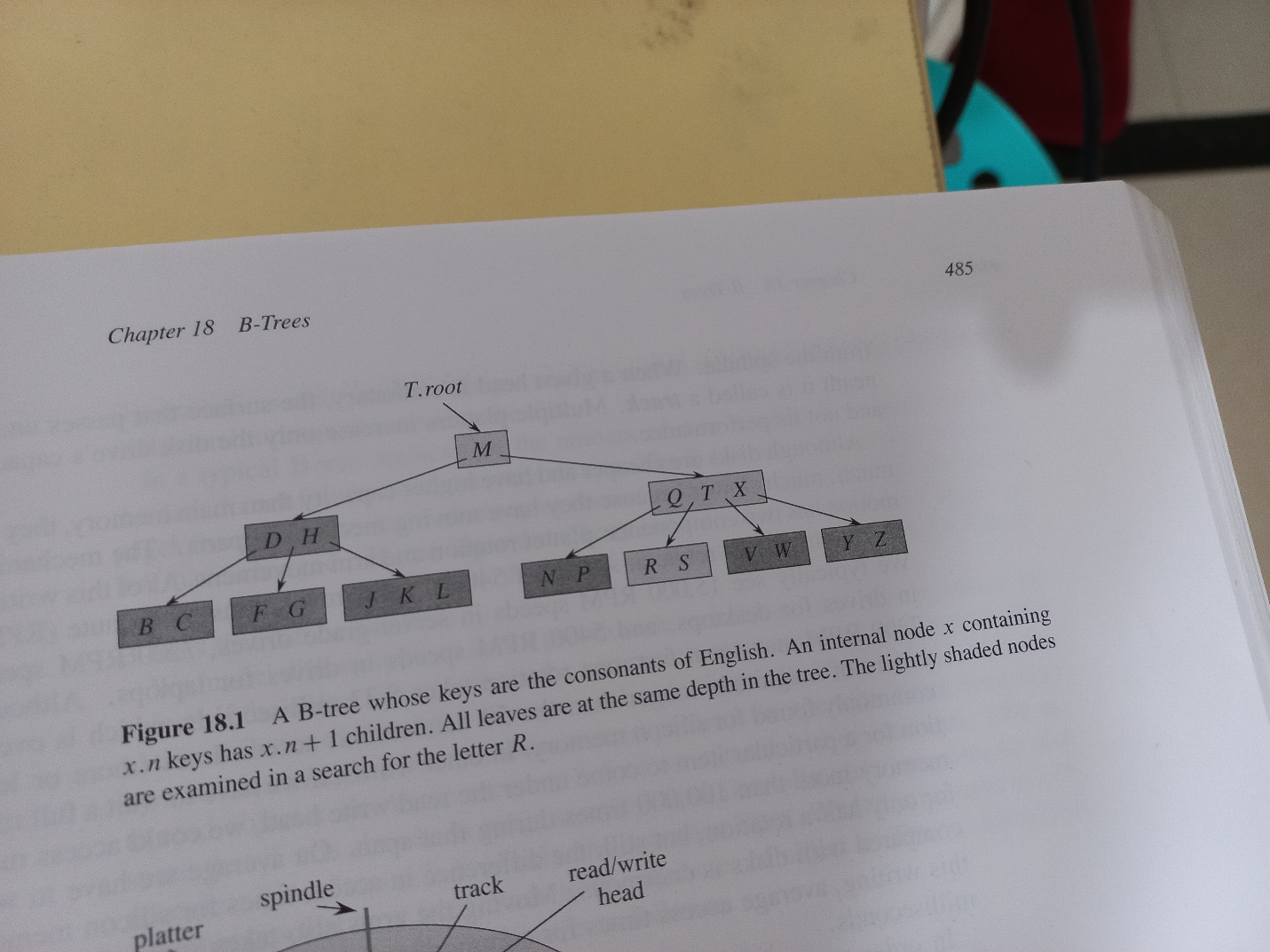
[property]

A B-Tree T is a rooted tree having the following properties.

1. Keys are stored in nondecreasing order.
2. And the keys separate the ranges of keys stored in each subtree.
3. All leaf nodes have same depth.
4. Nodes have upper bounds and lower bounds on the number of keys they contain.
5. For every nodes other than root node x, it must have at least t-1 keys and at most 2t-1 keys.
6. For root node, it must have at least 1 key iff the tree is nonempty.

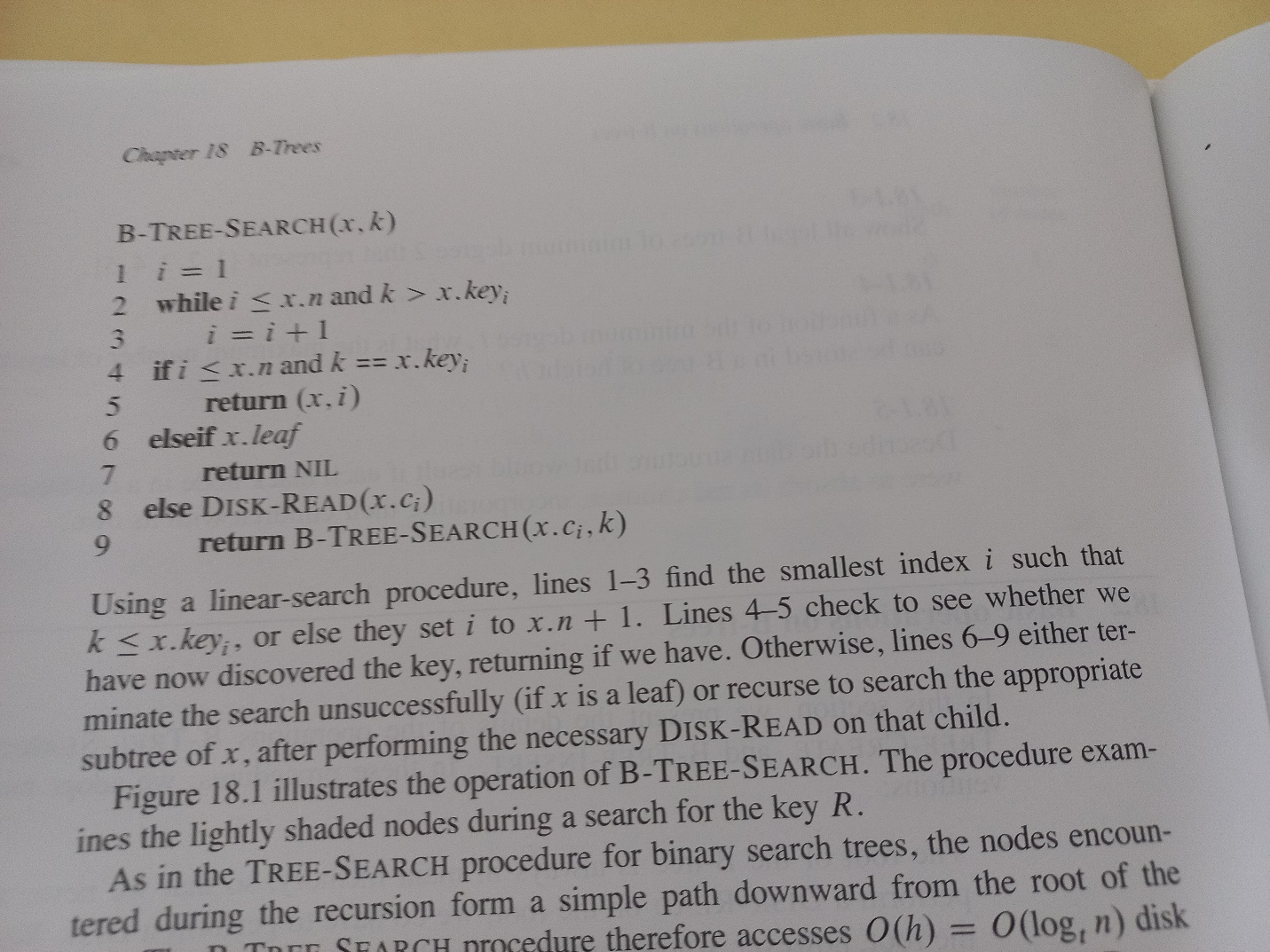
[terms]

1. depth (height): above
2. minimum degree: to express these bounds.
3. Full: the B-tree is full iff it contains exactly keys.

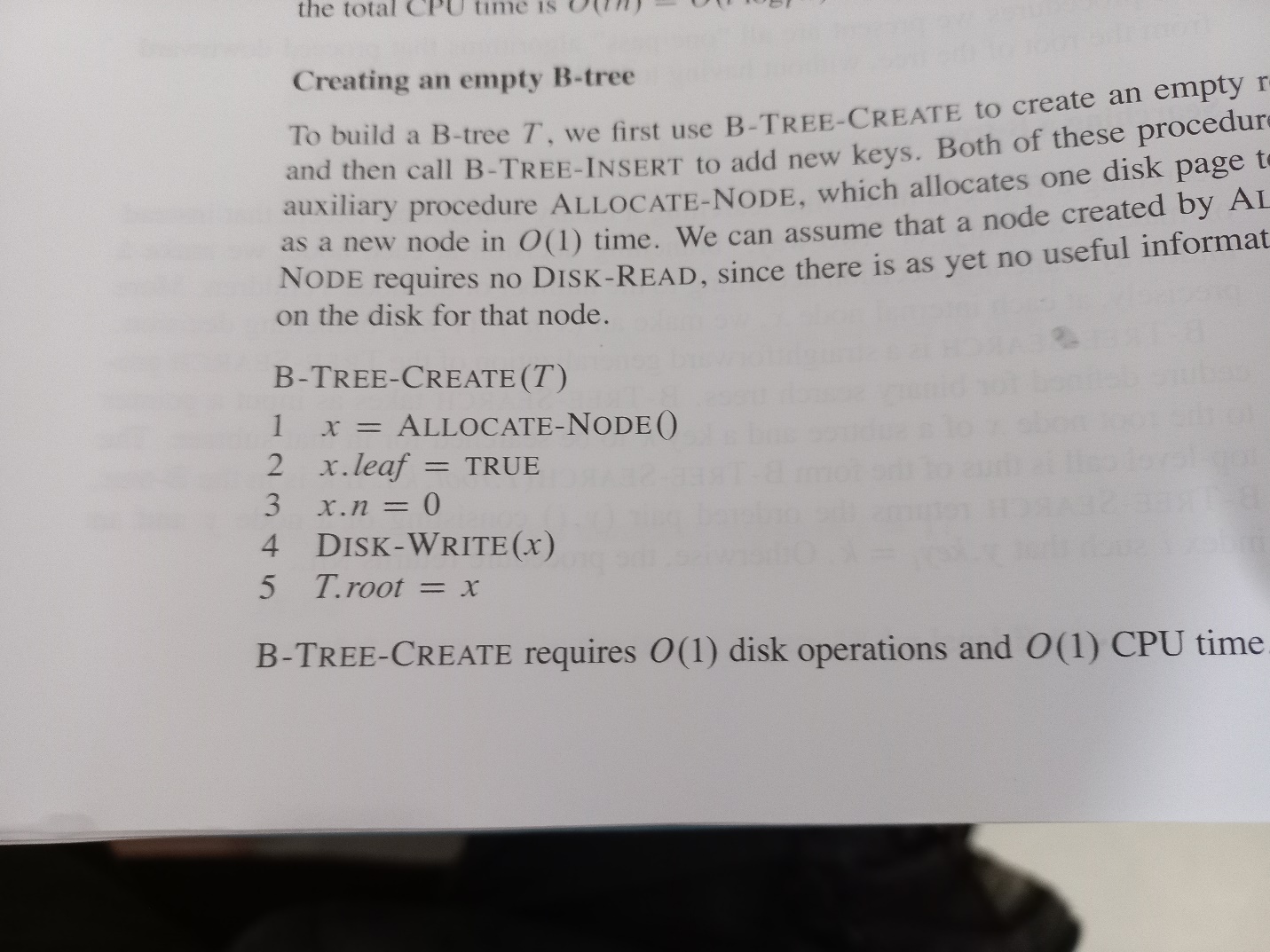


[operation]

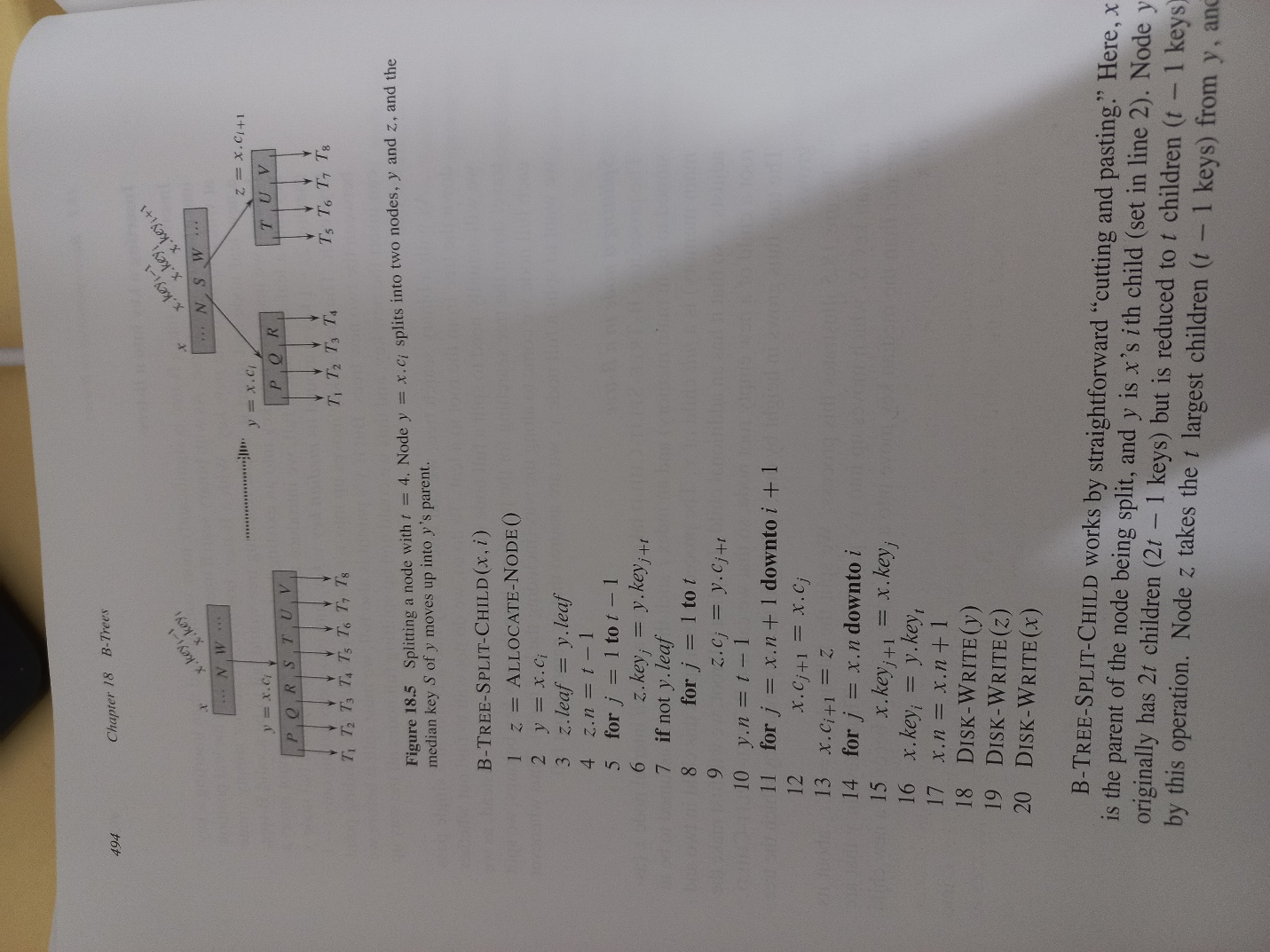
1. search: search for specified key.



1. create: create an empty B-tree given a tree.



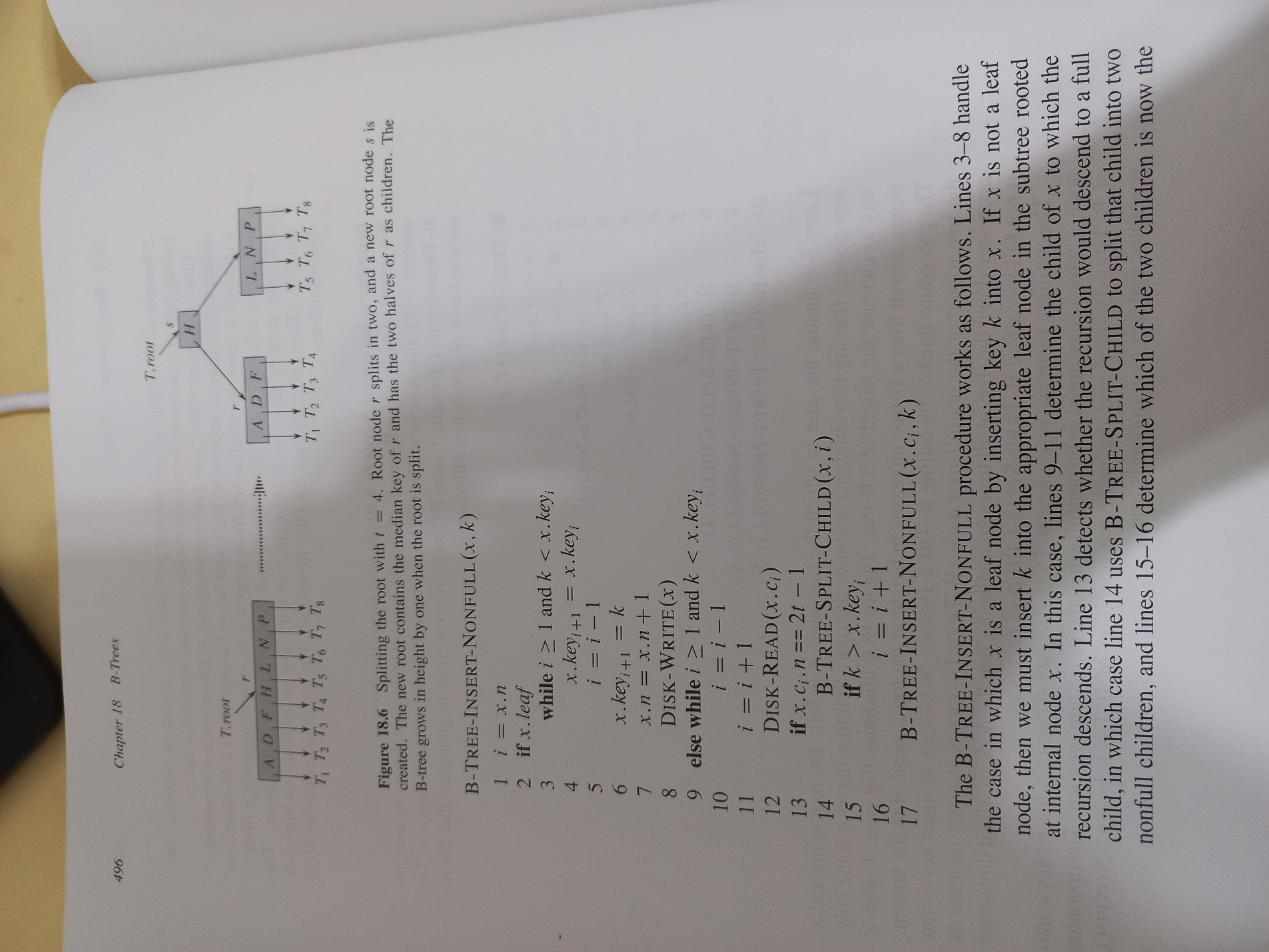
1. split: split a node consists of keys which has a parent into two nodes.



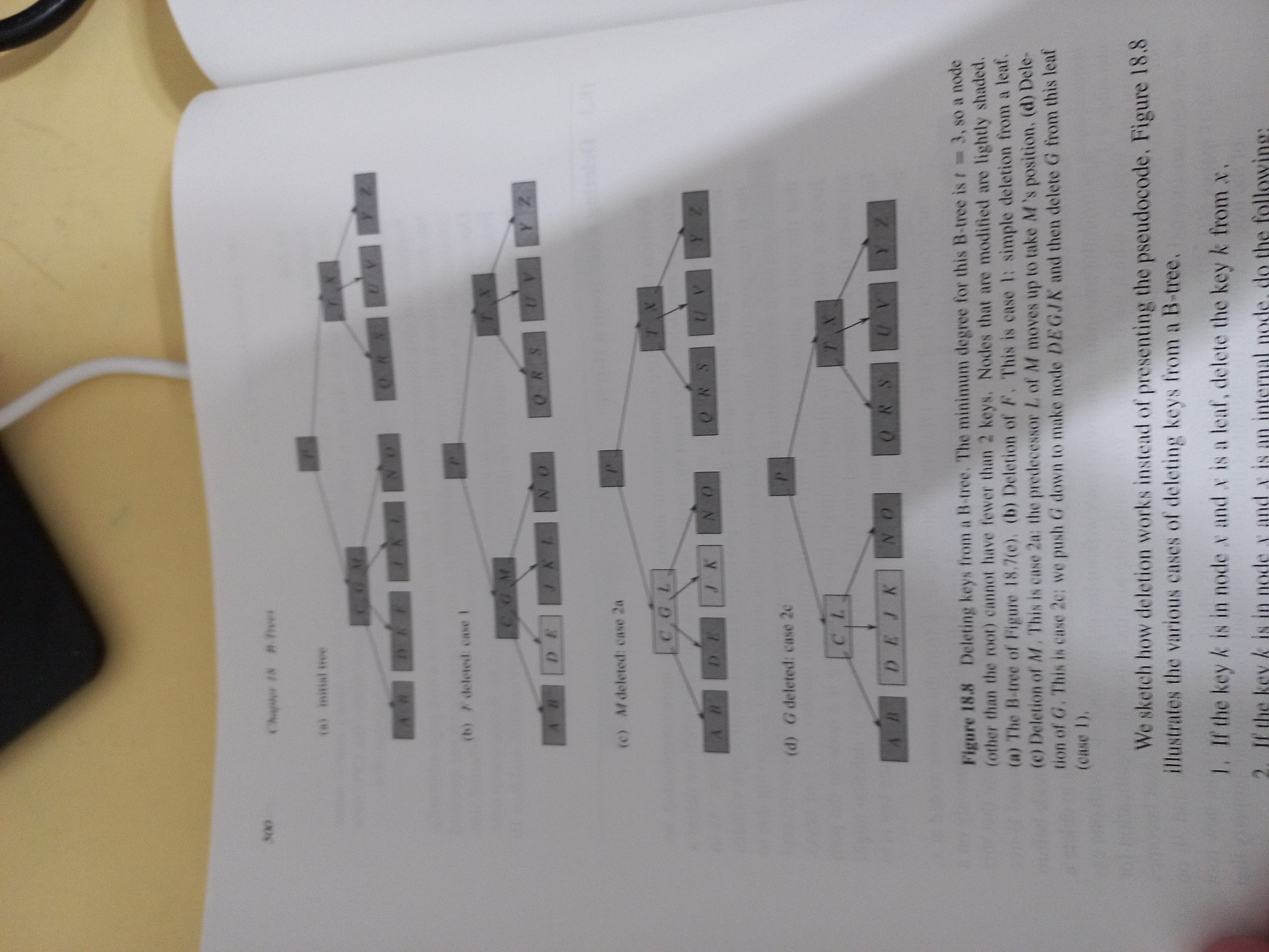
1. insert: insert a key in a single pass down the tree.



1. insert: split a node consists of keys which has NO parent into two nodes.



1. Deletion:



[theorem]

[pf]

