



# Solution

---

*NDBI007: Practical class 5*

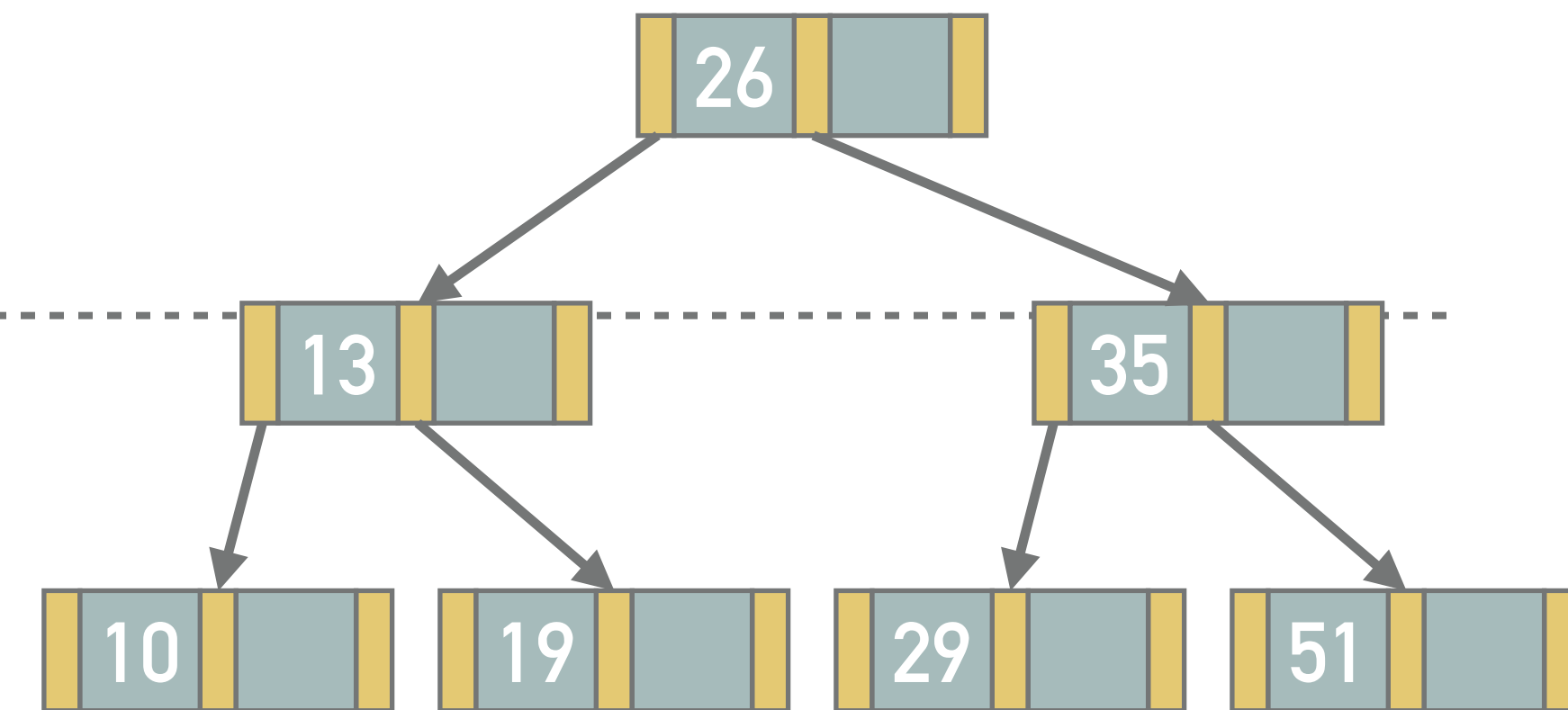




# Exercise 5.6 (Solution)

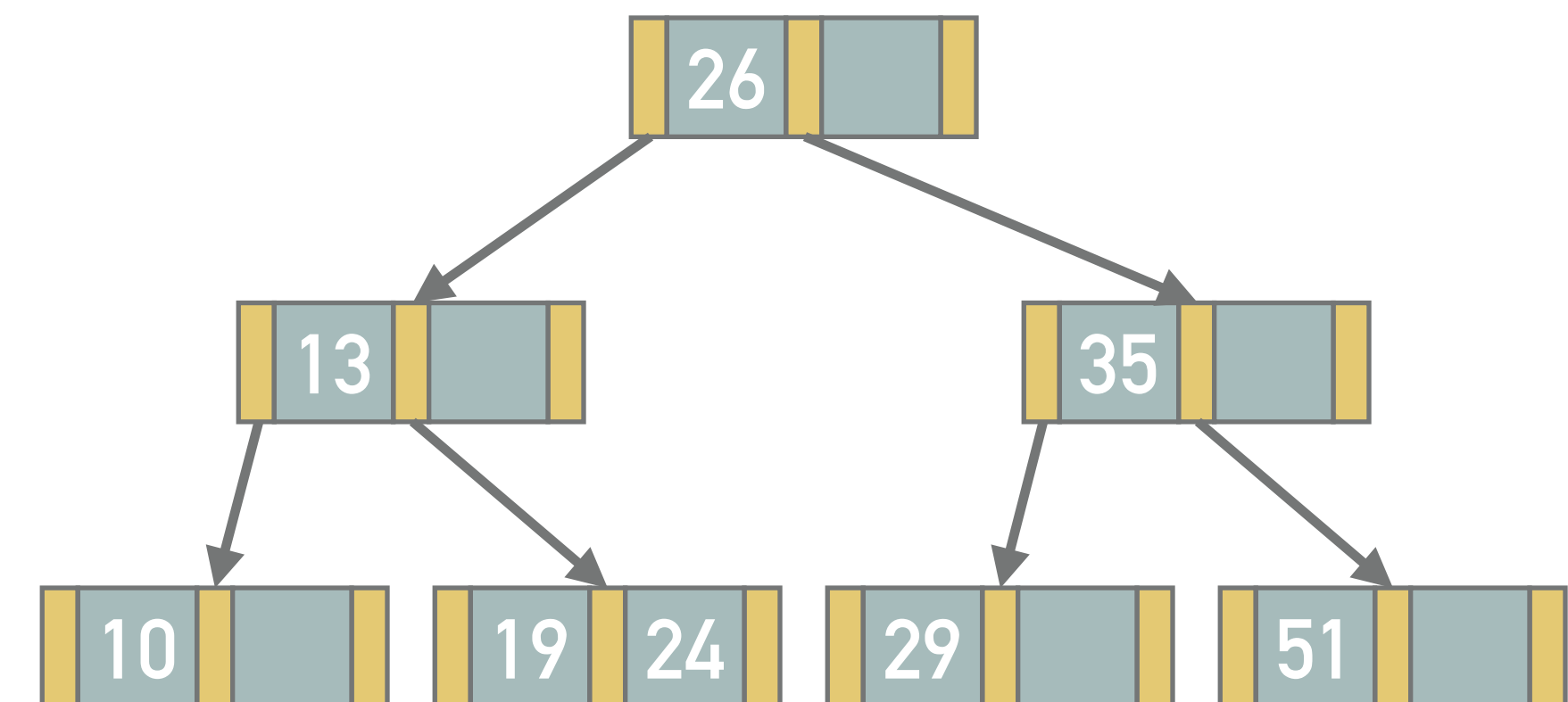
## ❖ Insertion of key 13

- ❖ Key 13 is less than 26, i.e., we navigate to the left (leaf) node (10, 19)
- ❖ After an attempt to insertion, (10, 13, 19) contains too many keys, a splitting occurs
- ❖ Key 13 goes to the parent (13, 26, 35) that also contains too many keys, i.e., additional splitting occurs and the tree height is increased



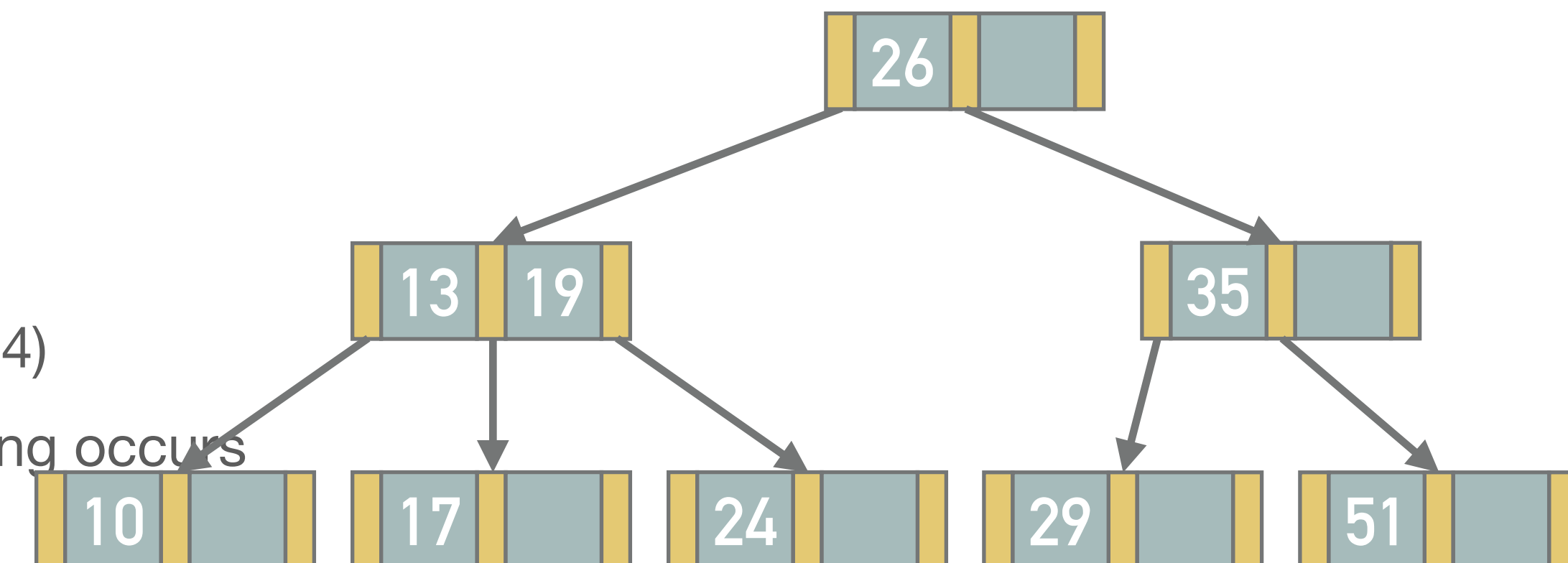
## ❖ Insertion of key 24

- ❖ Key 24 is less than 26, i.e., we navigate to the left node (13)
- ❖ Key 24 is greater than 13, i.e., we navigate to the right node (19)
- ❖ Key 24 goes to the leaf (19, 24)



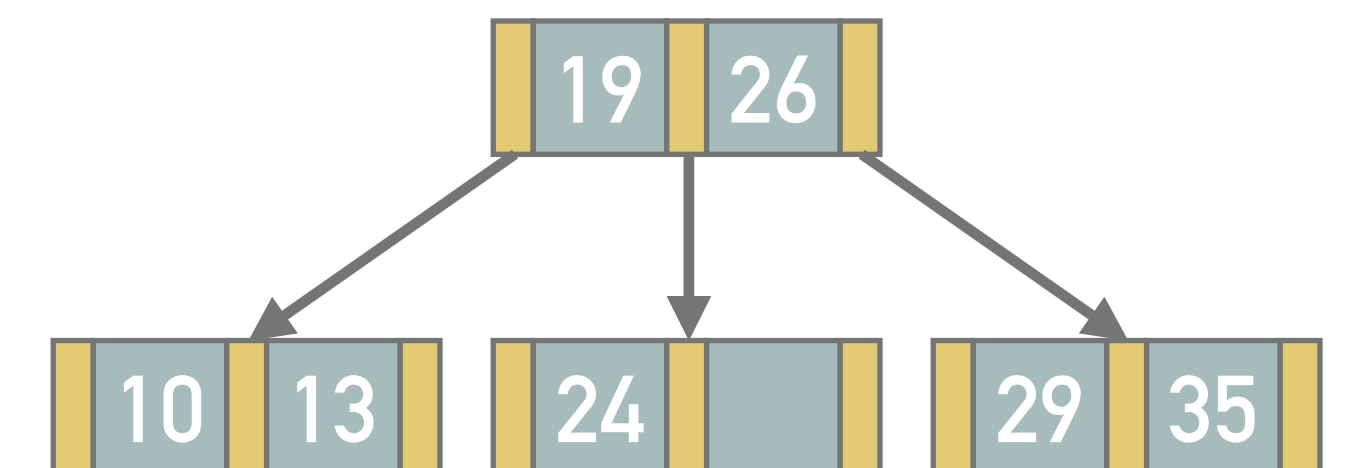
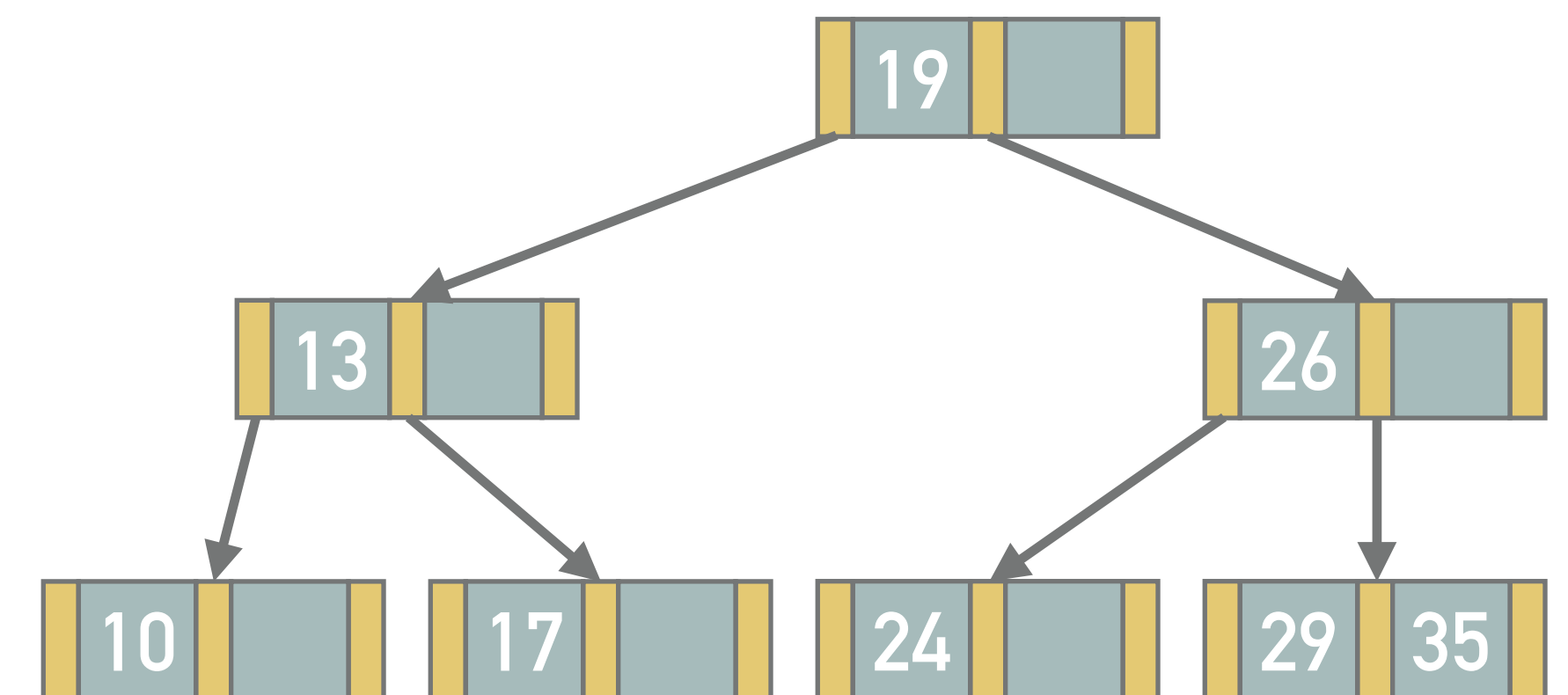
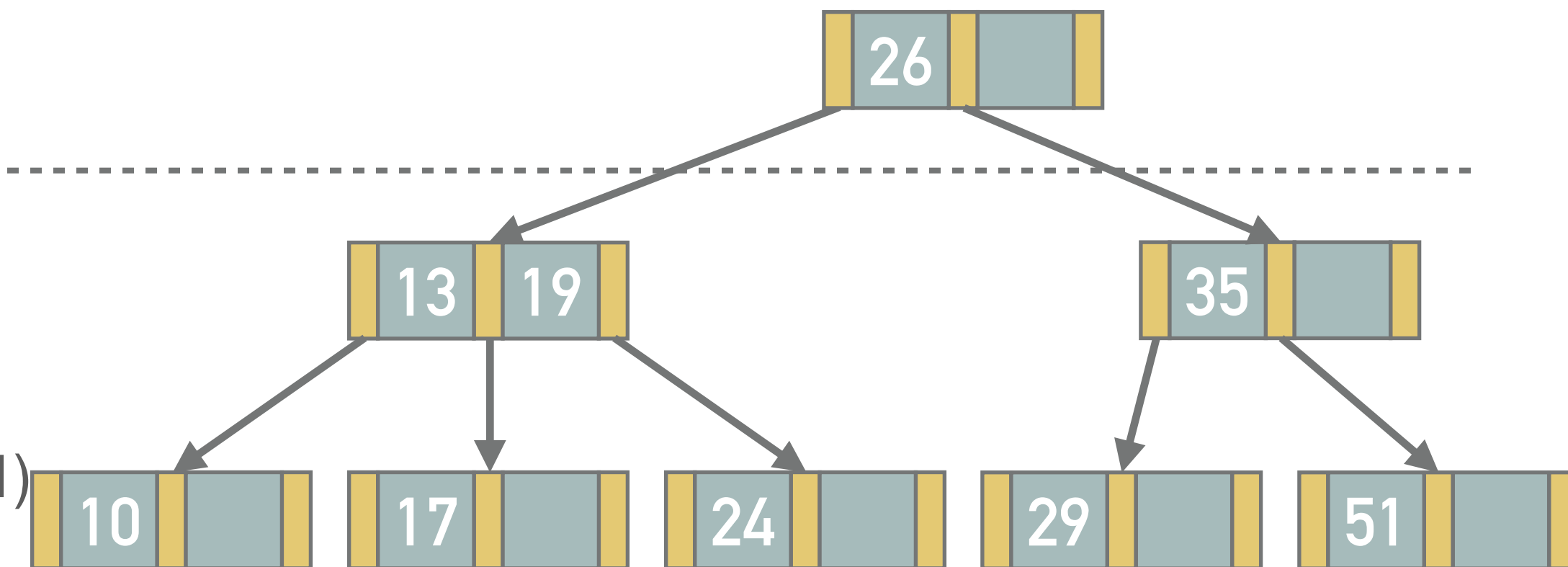
## ❖ Insertion of 17

- ❖ Key 17 is less than 26, i.e., we navigate to the left node (13)
- ❖ Key 17 is greater than 13, i.e., we navigate to the right (leaf) node (19, 24)
- ❖ After the insertion, the leaf (17, 19, 24) contains too many keys, a splitting occurs
- ❖ Key 19 goes to the parent node (13, 19)



# Exercise 5.6 (Solution Continued)

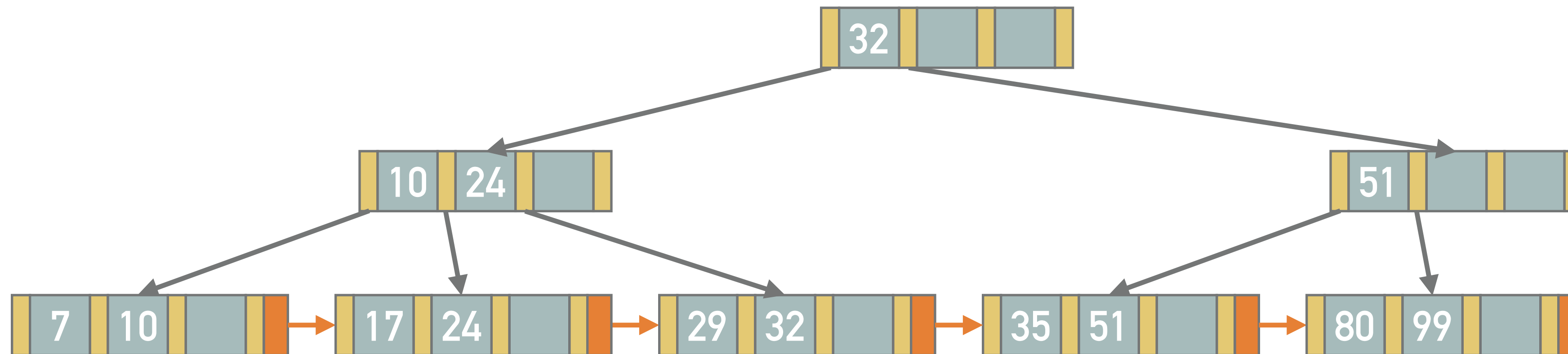
- ❖ Deletion of record with key 51
  - ❖ Key 51 is greater than 26, i.e., we navigate to the right node (35)
  - ❖ Key 51 is greater than 35, i.e., we navigate to the right (leaf) node (51)
  - ❖ Key 51 is removed, but we have to merge nodes (empty) and (35)
  - ❖ Parent node (empty) has too few keys, therefore we take key from the left sibling (26)
  - ❖ Finally, taking a key from the left children (13, 19), 19 goes to the root node
- ❖ Deletion of record with key 17
  - ❖ Key 17 is less than 19, i.e., we navigate to the left node (13)
  - ❖ Key 17 is greater than 13, i.e., we navigate to the right (leaf) node (17)
  - ❖ Key 17 is removed, causing the splitting of nodes (10) and (13)
  - ❖ Finally, we have to merge nodes (19, 26) to establish a new root node



# Exercise 5.12 (Solution)

---

- ❖ The insertion of a record with key 51 splits the right lead node into nodes (29, 32) and (35, 51)
  - ❖ The separating value 32 is inserted into the parent node where there is enough space
- ❖ The insertion of a key 80 is trivial, it goes to the node (35, 51, 80)
- ❖ The insertion of a key 99 splits the right leaf node into nodes (35, 51) and (80, 99)
  - ❖ The separating value 51 is inserted into the parent node (10, 24, 32, 51) where it leads to the cascade split
  - ❖ The parent is split to nodes (10, 24) and (51) while (32) goes to the root node



## Exercise 5.17 (Solution)

---

- ❖ We can delete the record 29 while moving the record 32 to the neighboring node and modification of the split value
- ❖ The record with key 19 will be removed while moving the record with key 17 and modification of the split value
- ❖ The record with key 17 will be removed while moving the record with key 13 and modification of the split value
- ❖ Removing another record would lead to merging of the tree nodes into two

