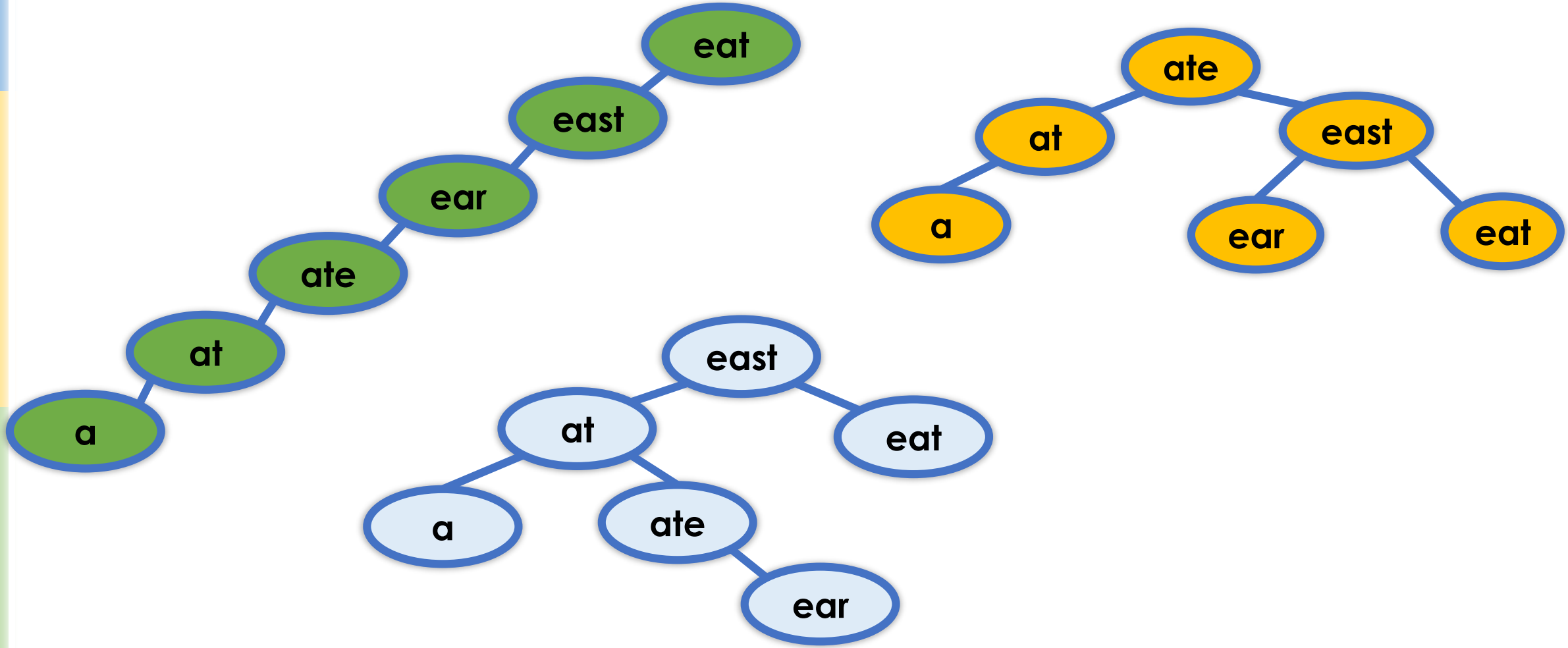
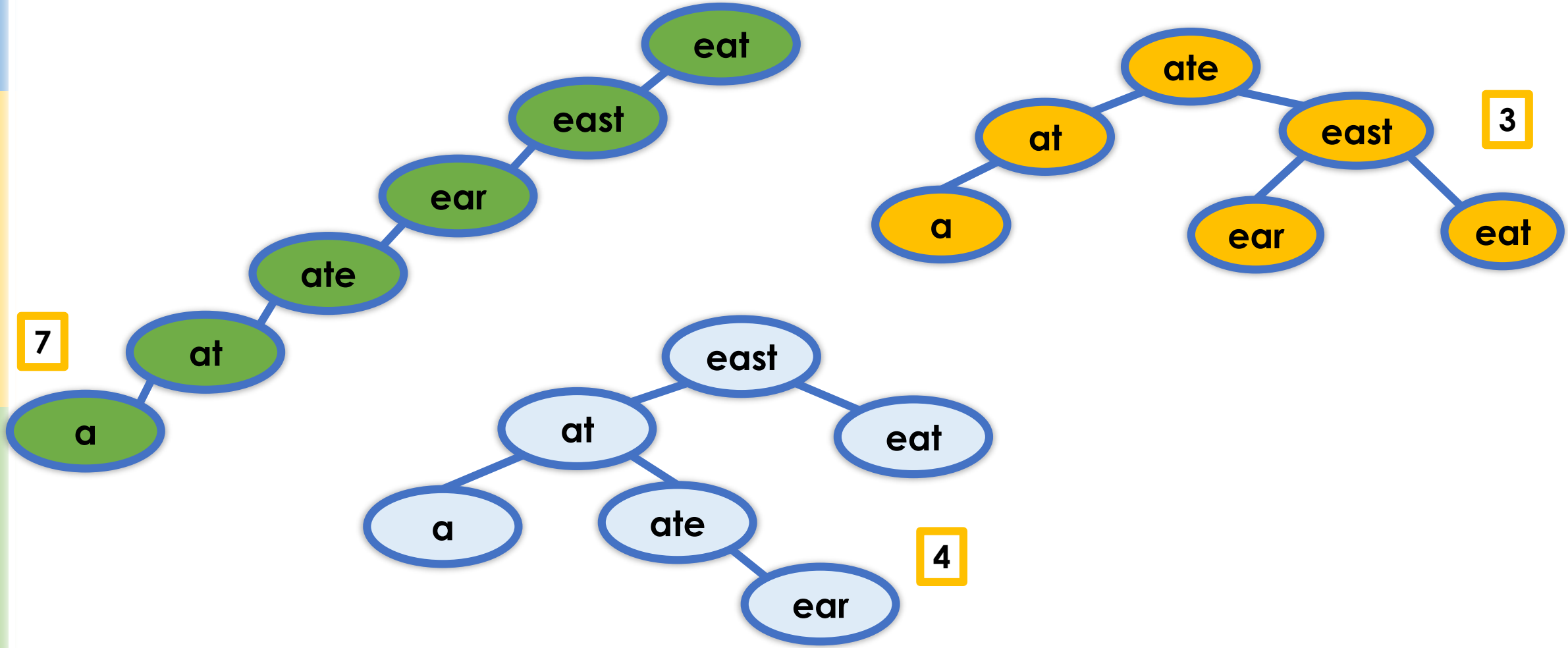
Binary Search Trees

Performance

Max distance until leaf?

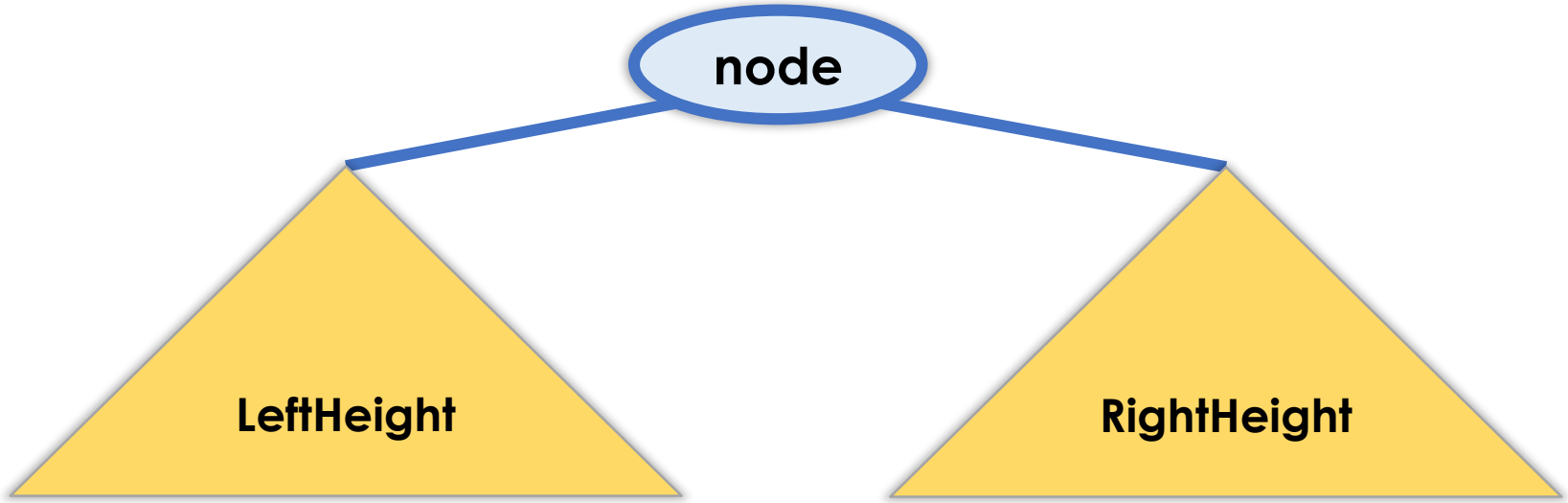


Max distance until leaf?

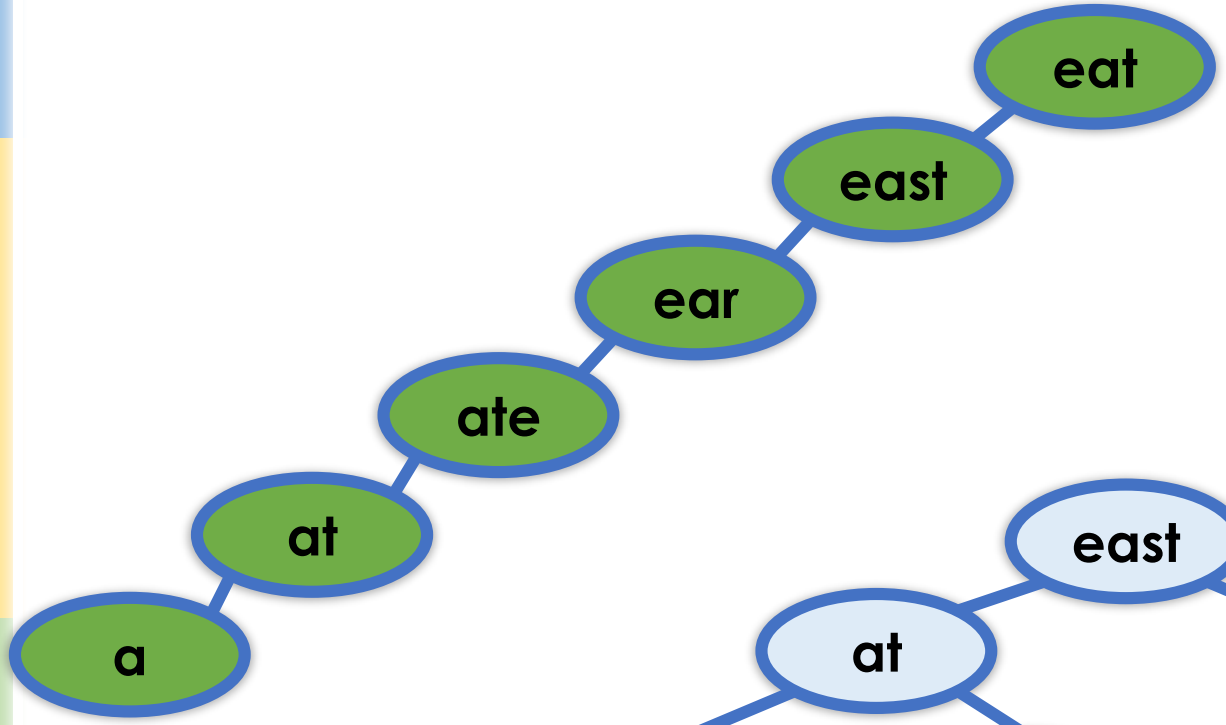


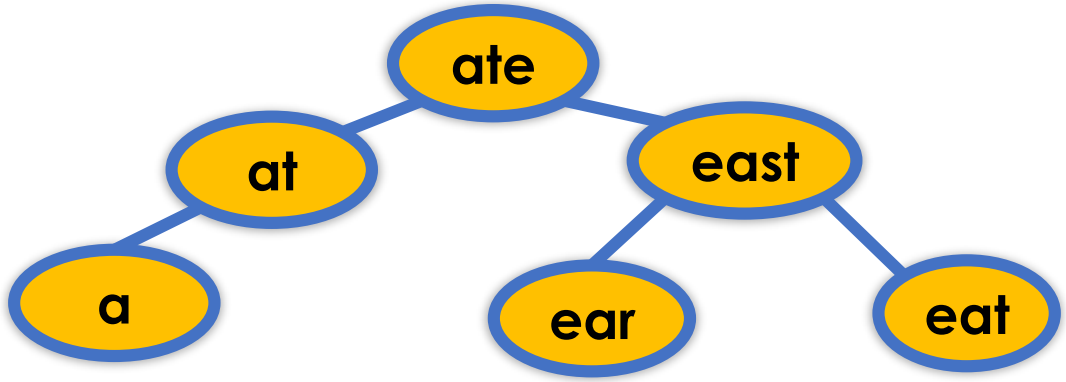
Balanced BST

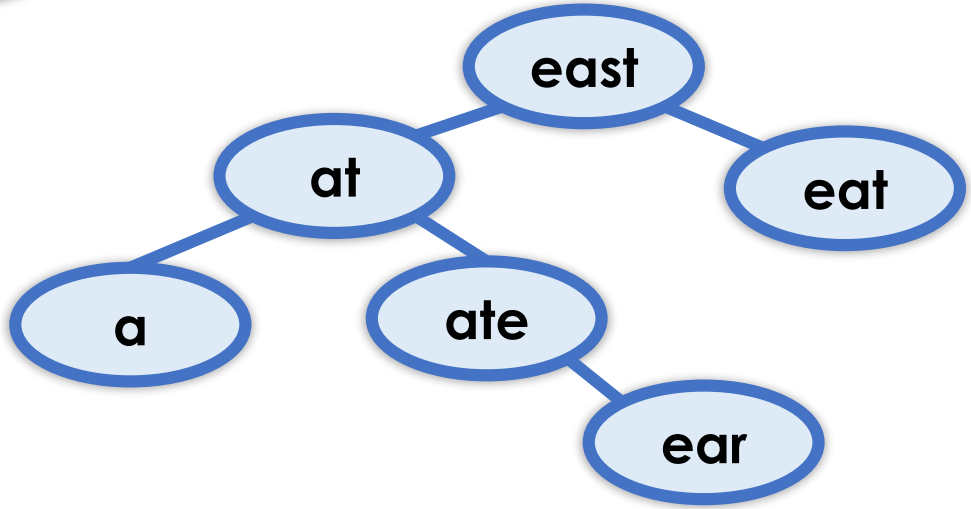
| LeftHeight – RightHeight | <=1



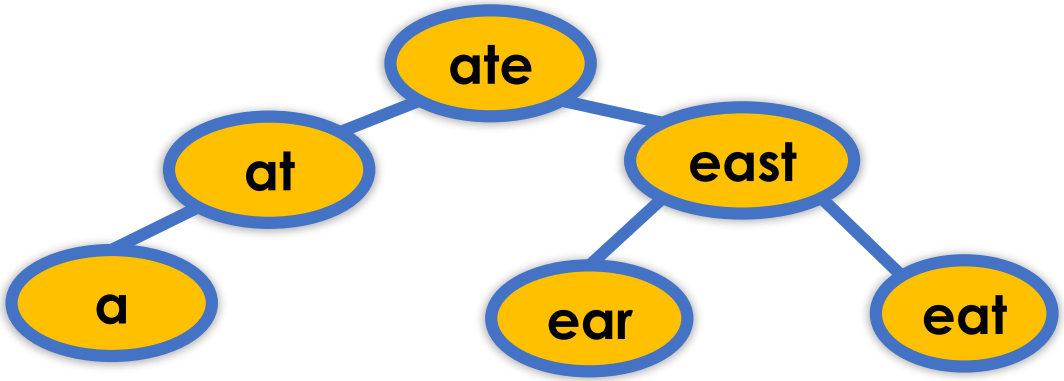
Balanced BST IVQ: which are?





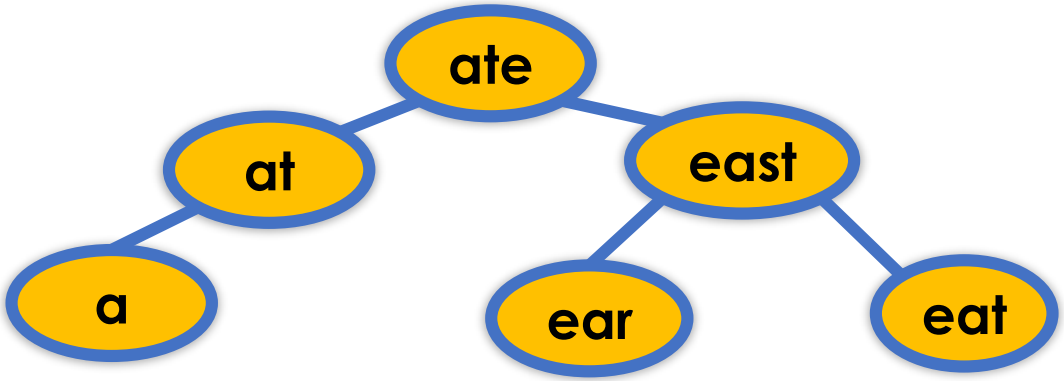


Balanced BST



Balanced BST

height ≈ log(n)



`isWord(String wordToFind)`

	Best case	Average case	Worst case
Linked List	**O(1)**	**O(n)**	**O(n)**
BST	**O(1)**	**O(log n)**	**O(n)**
Balanced BST	**O(1)**	**O(log n)**	**O(log n)**

`isWord(String wordToFind)`

	Best case	Average case	Worst case
Linked List	**O(1)**	**O(n)**	**O(n)**
BST	**O(1)**	**O(log n)**	**O(n)***
Balanced BST	**O(1)**	**O(log n)**	**O(log n)**

*** Especially if insert to BST in order!**

`isWord(String wordToFind)`

	Best case	Average case	Worst case
Linked List	**O(1)**	**O(n)**	**O(n)**
BST	**O(1)**	**O(log n)**	**O(n)**
Balanced BST	**O(1)**	**O(log n)**	**O(log n)**

How to keep balanced? TreeSet in Java API

Thought question

- What's the performance of other operations?
  - **`isWord()`**
  - **`addWord()`**