

Trees

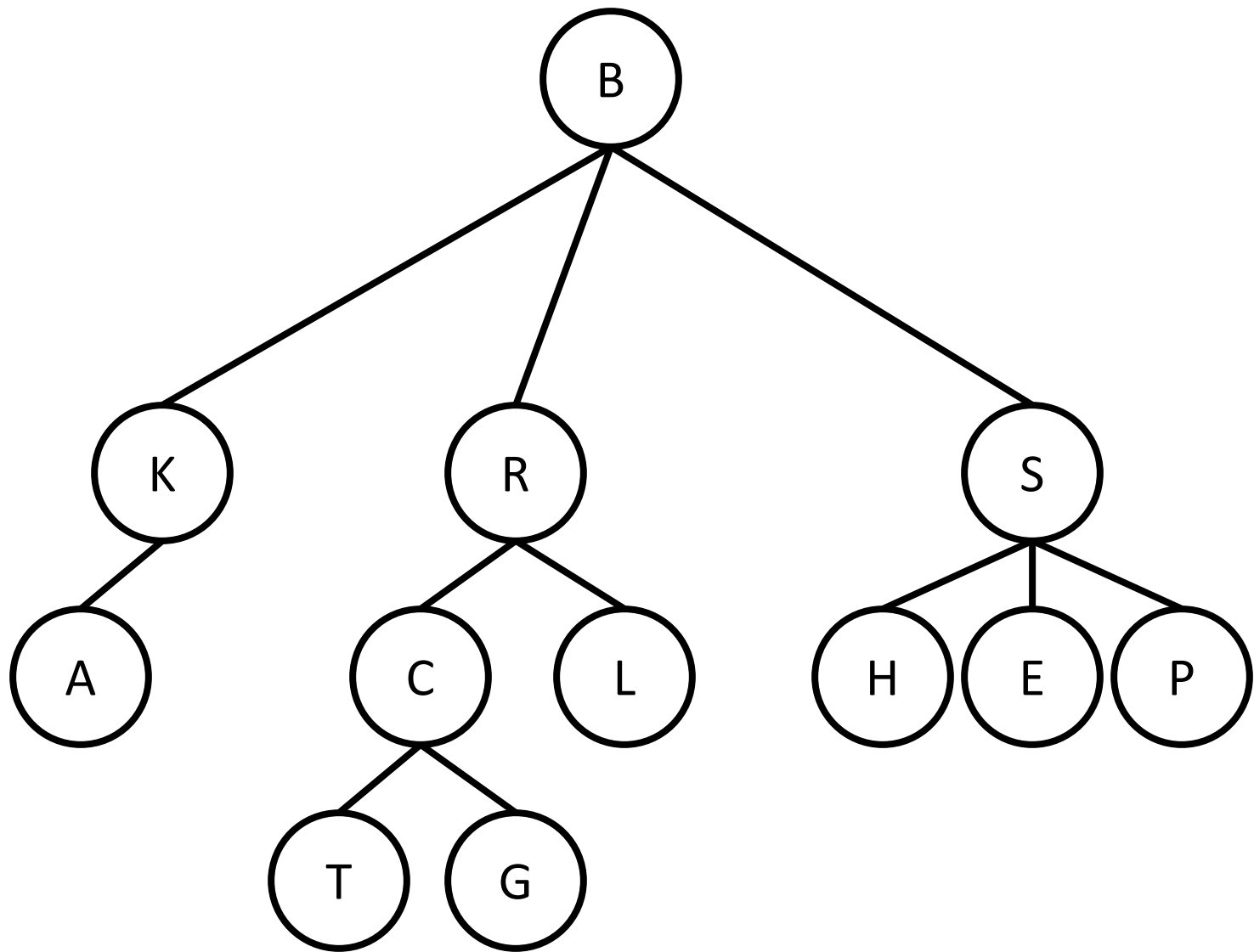


A New Math Type: *tree*

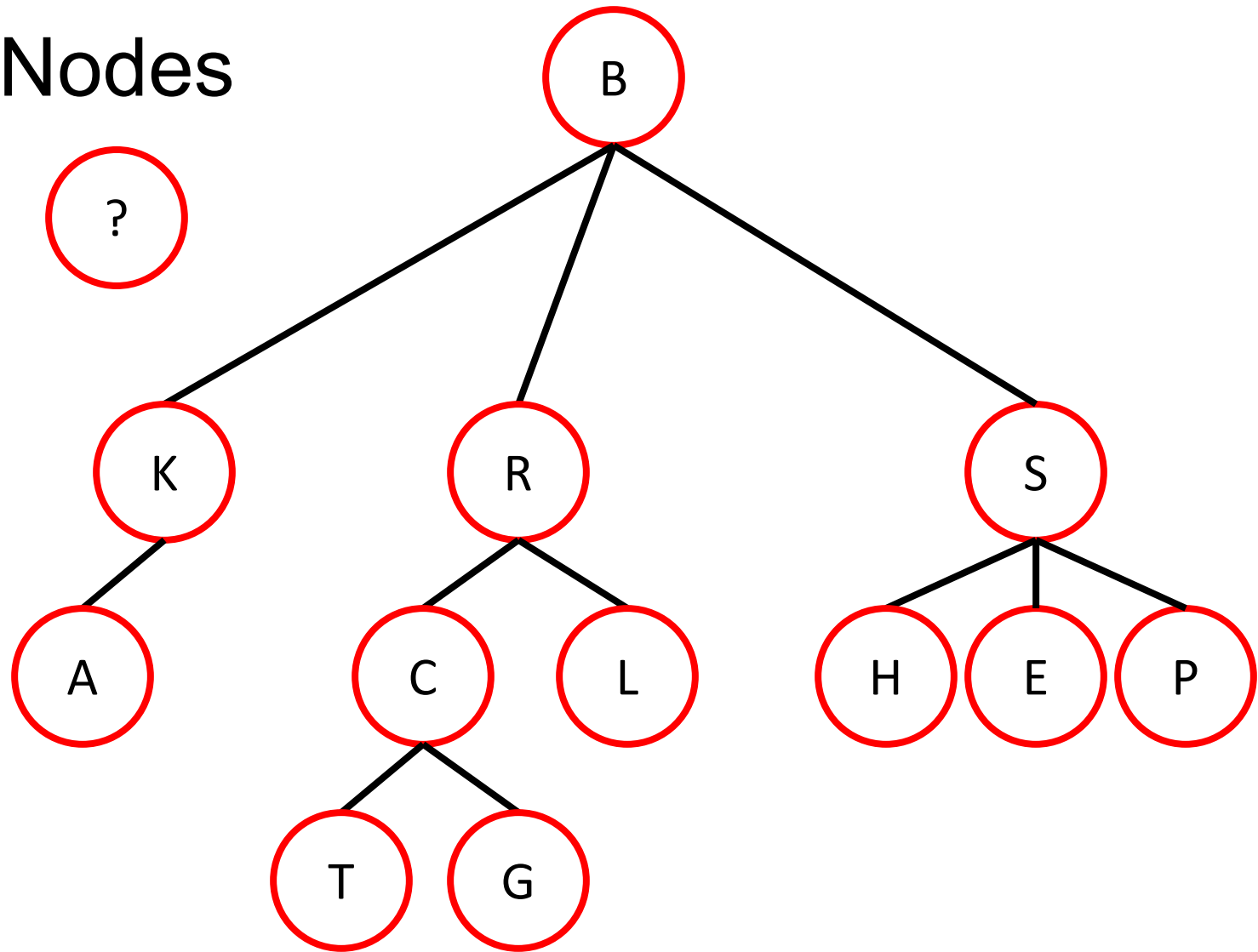
- A ubiquitous concept in computing is that of a *tree*
 - Often we are interested in a *binary tree*, a special case of a tree in which each *node* has at most two *children*
- An *informal* introduction (“node”?, “children”?) follows, using pictures rather than any new mathematical notation

Recursive Structure

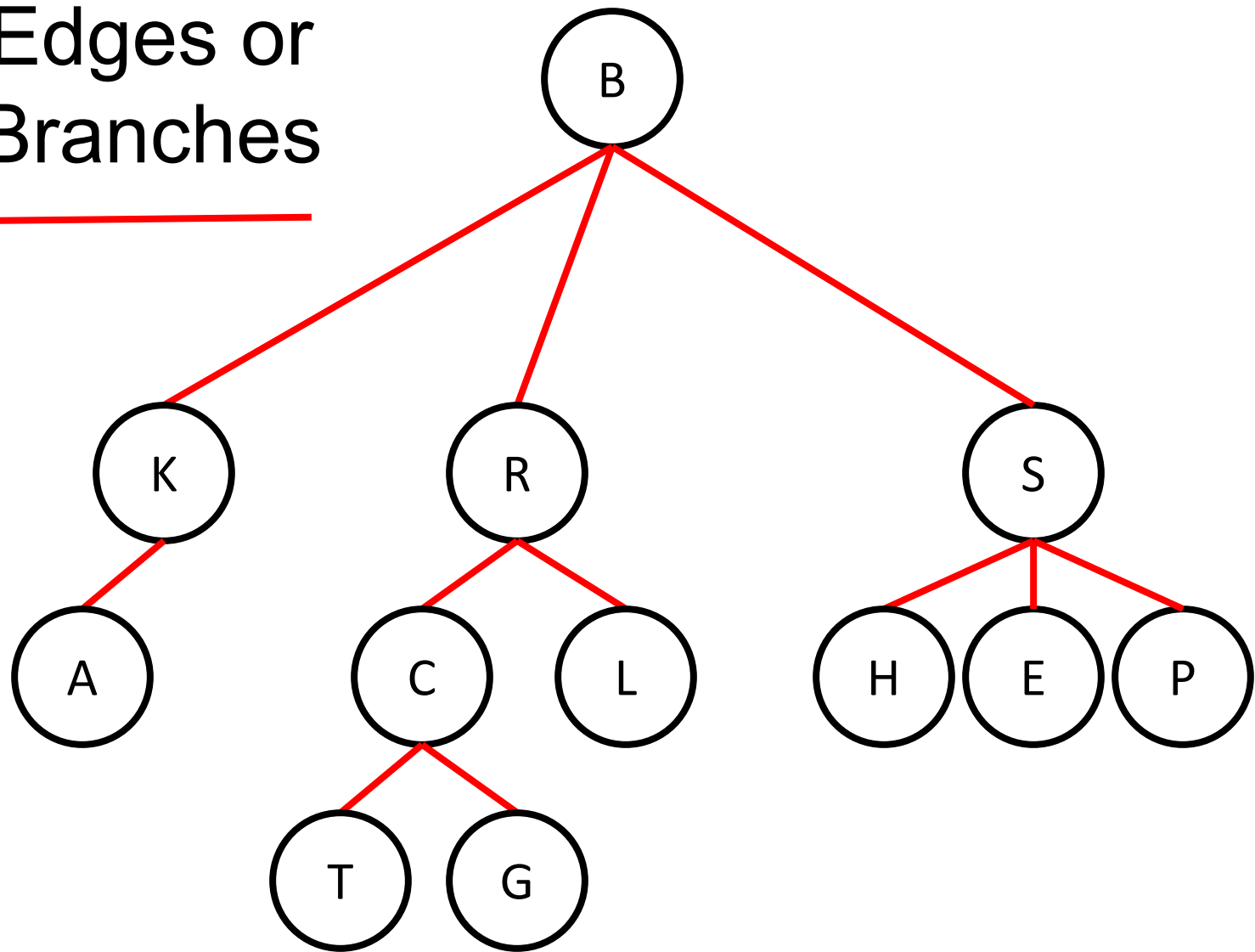
- A **tree** is made up of:
 - A **root node**
 - A string of zero or more **child nodes** of the root, each of which is the root of its own tree
- Since a tree may contain other trees, its structure is **recursive**
- Note: the following explanation of trees is adequate for present purposes but is not technically complete; details later...

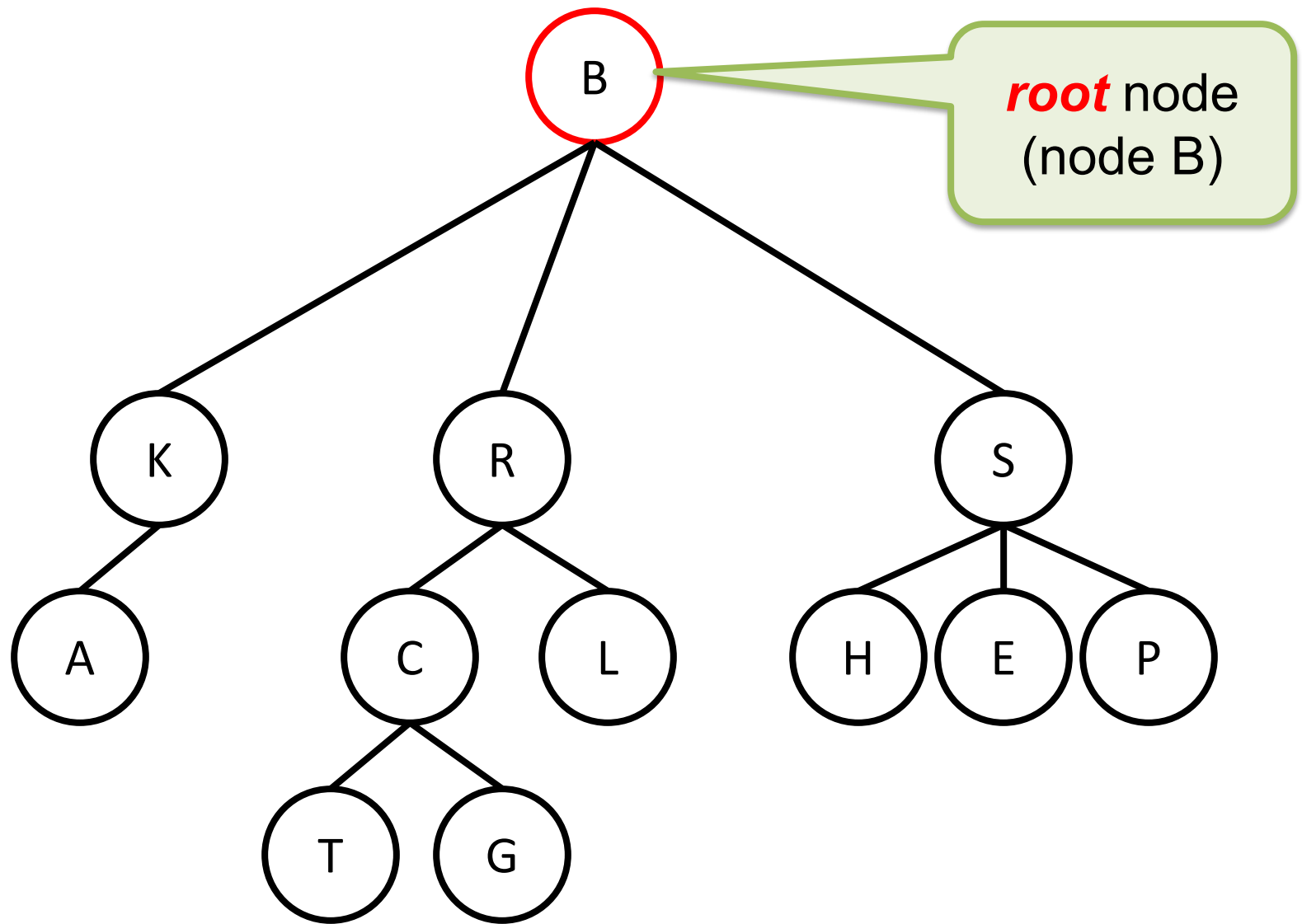


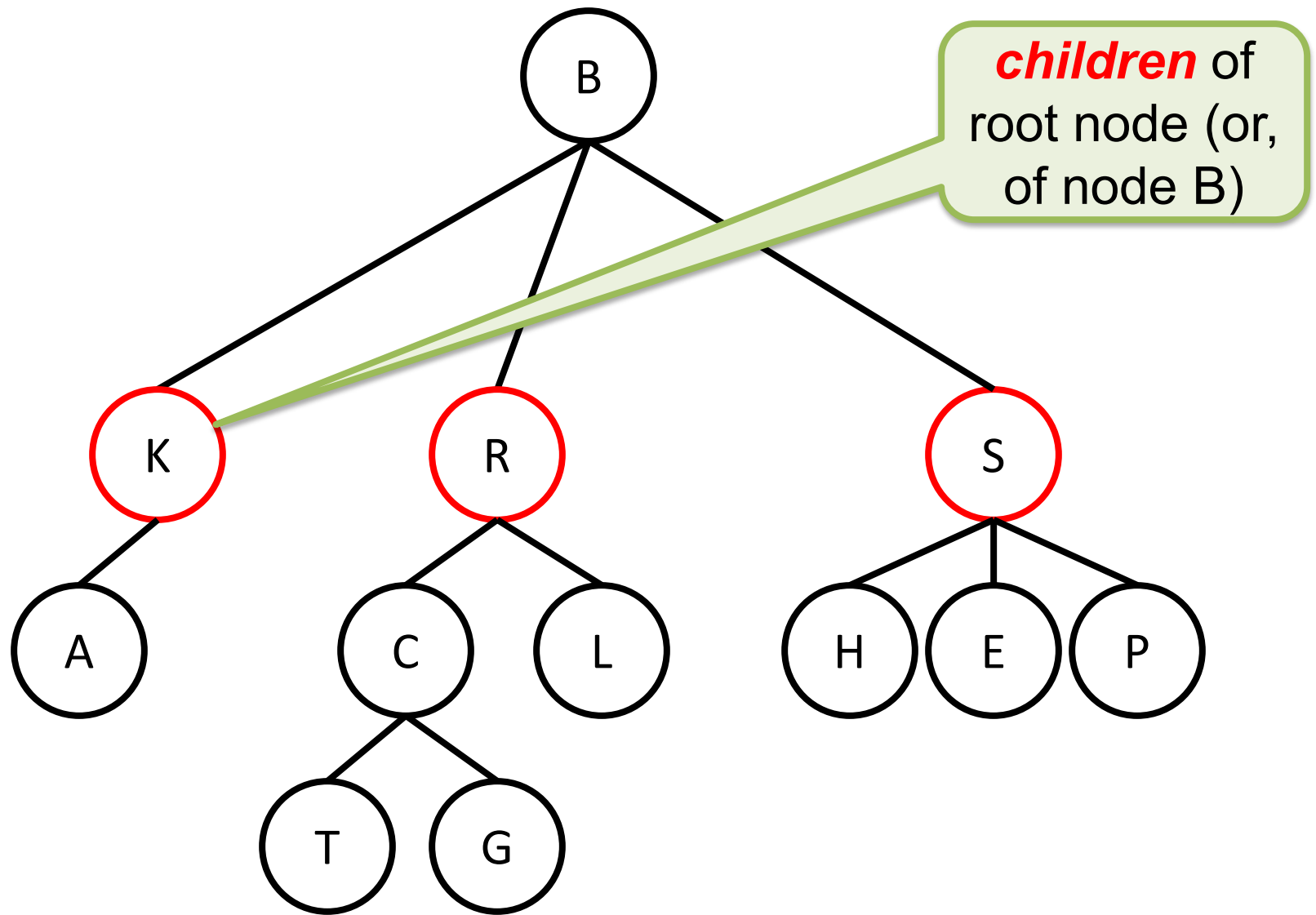
Nodes

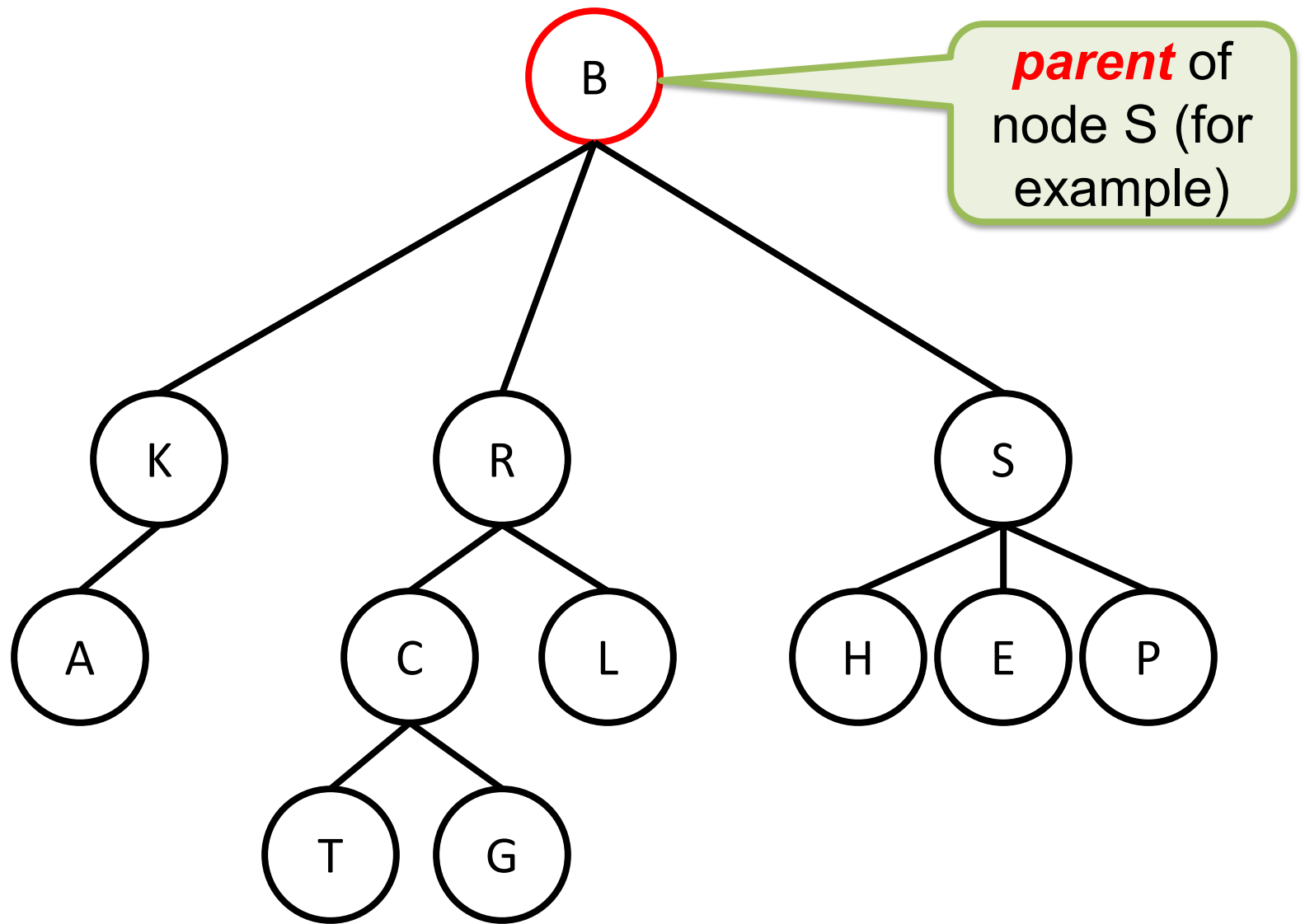


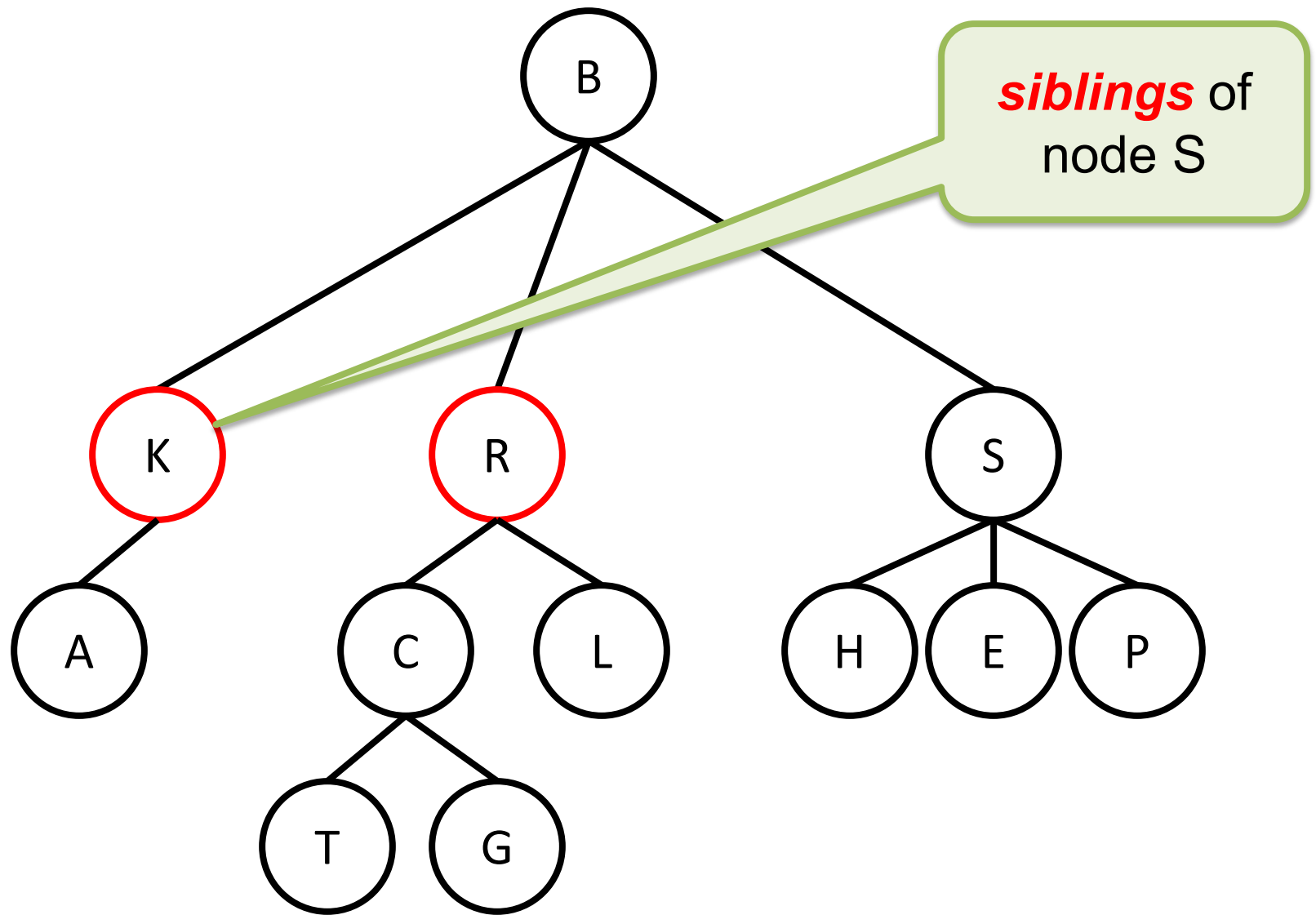
Edges or Branches

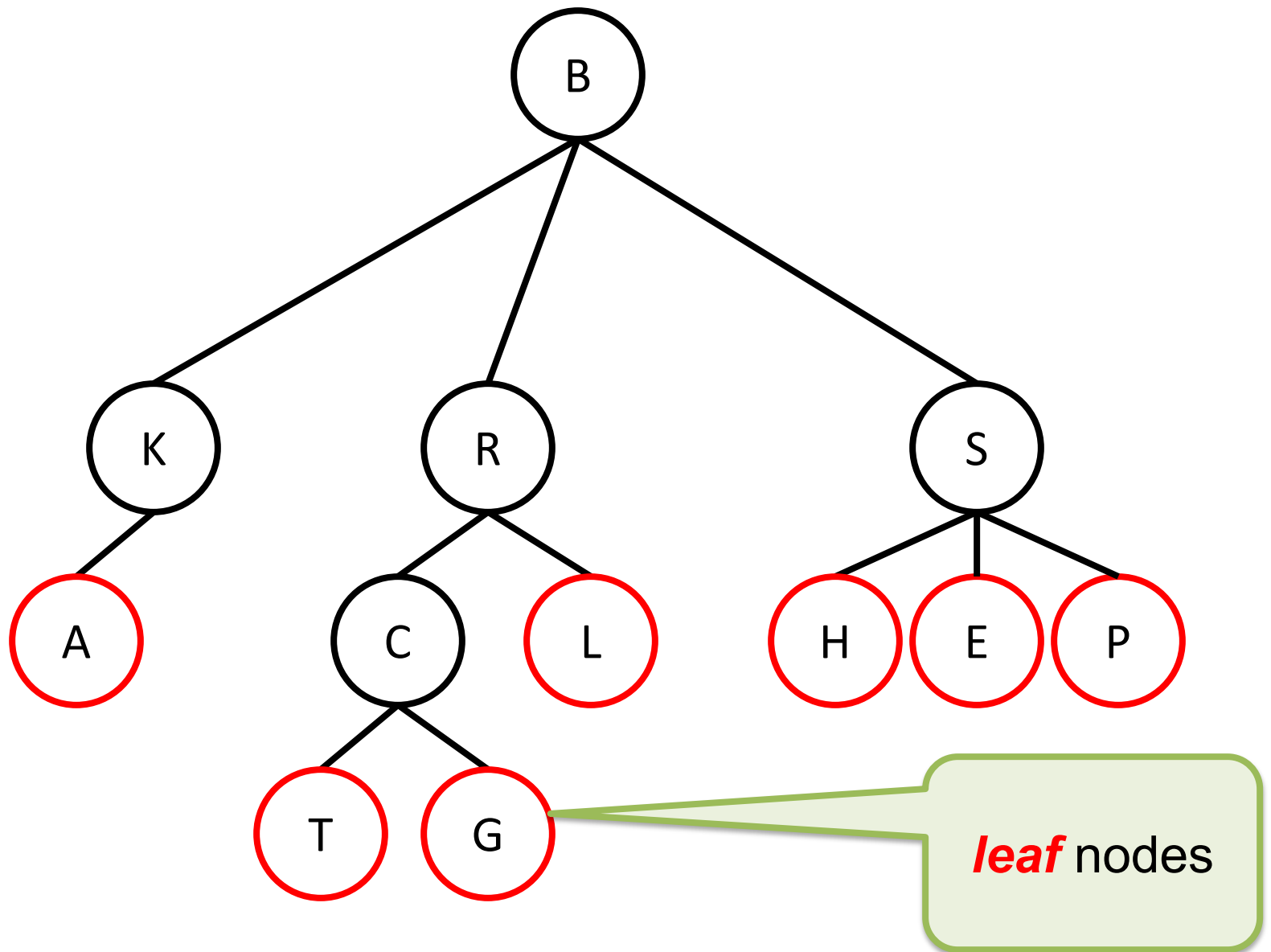




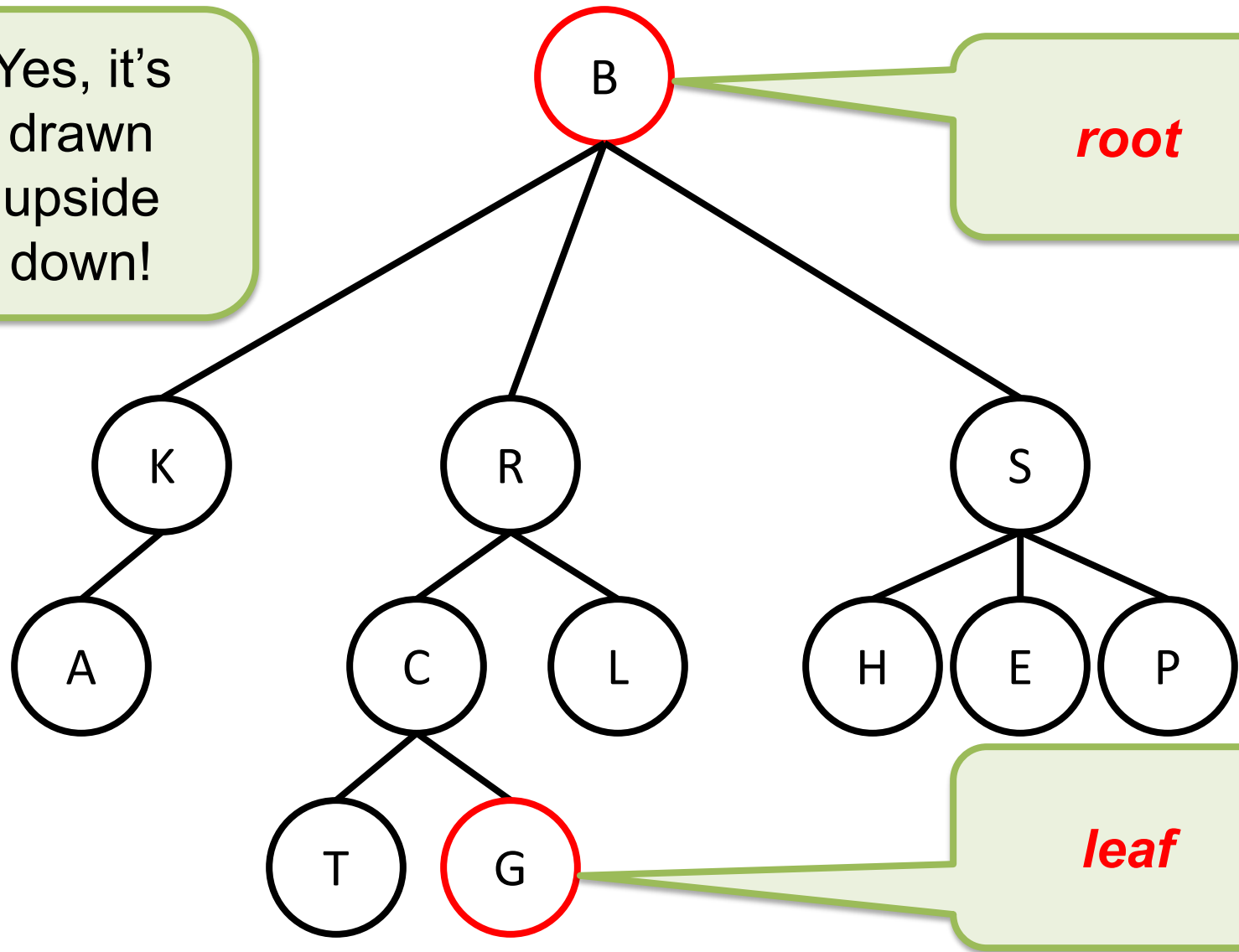


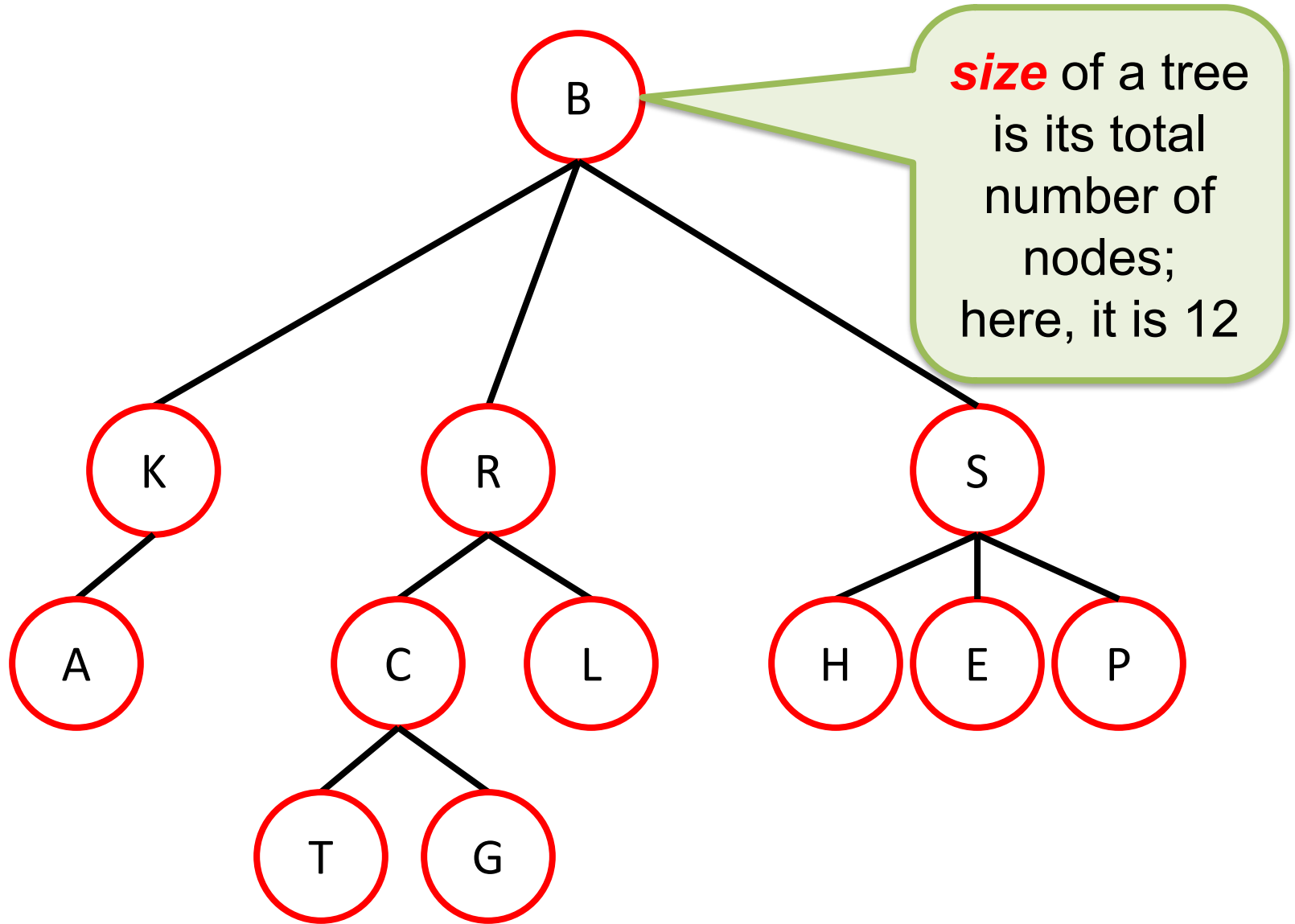






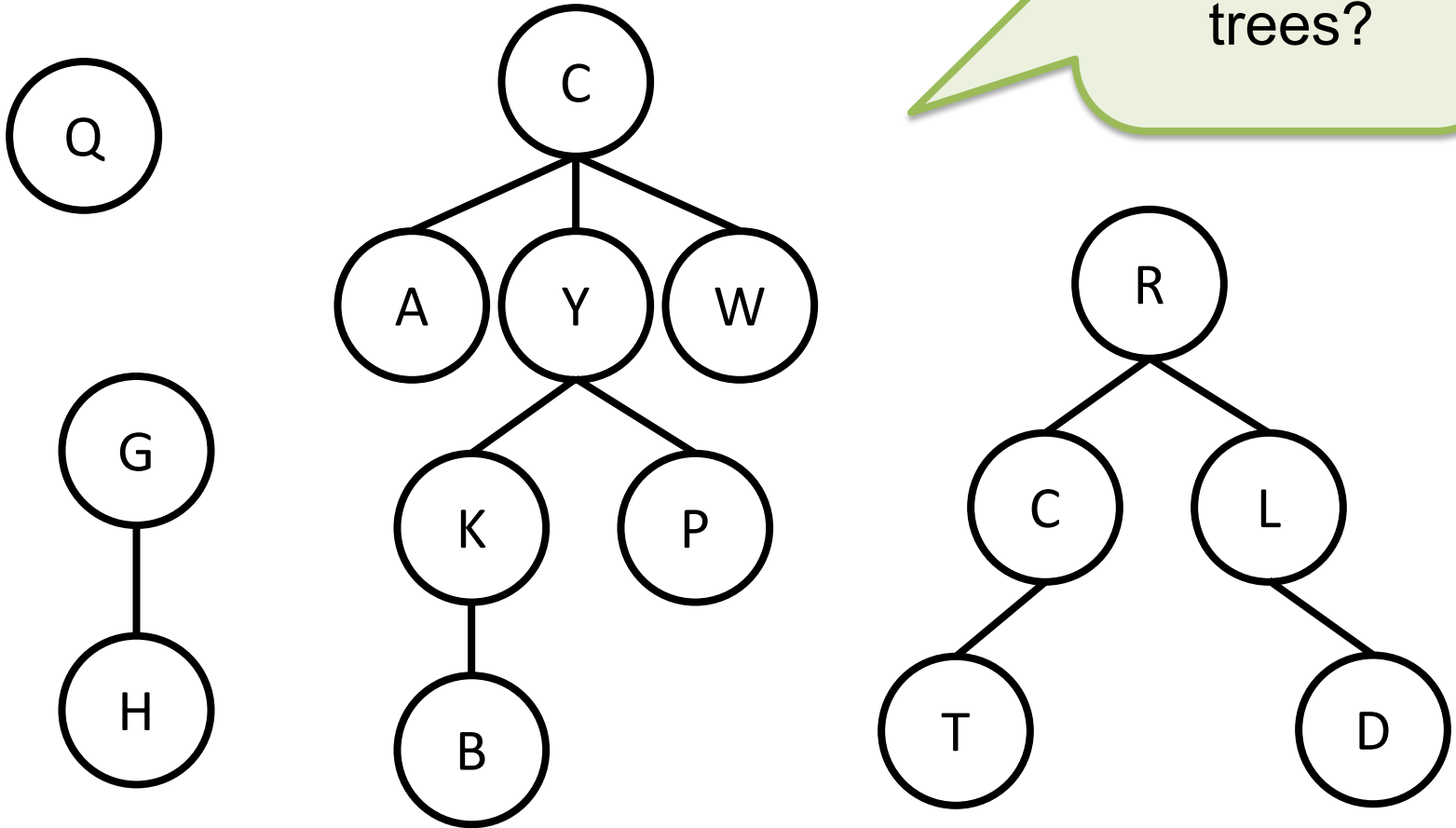
Yes, it's
drawn
upside
down!

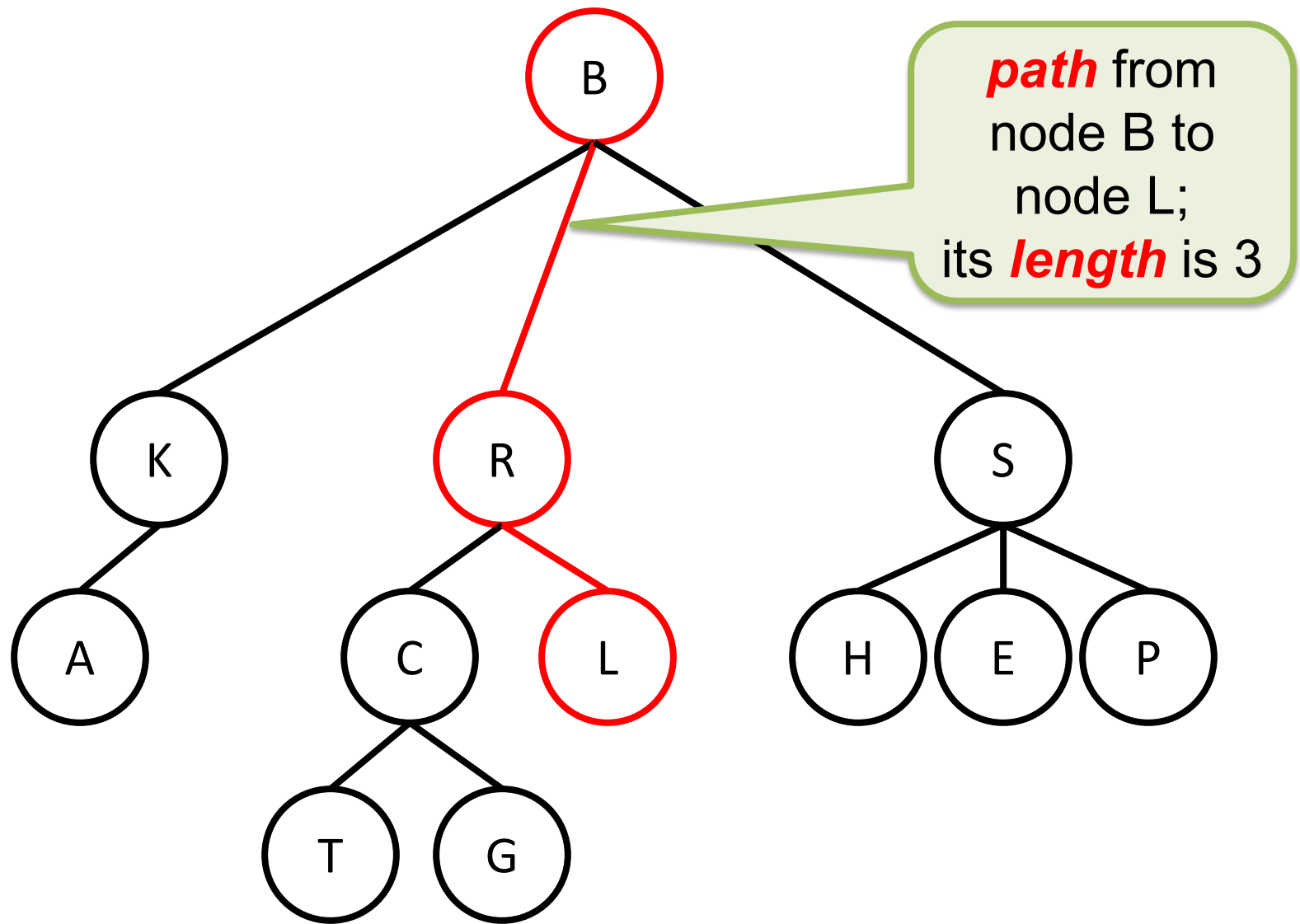


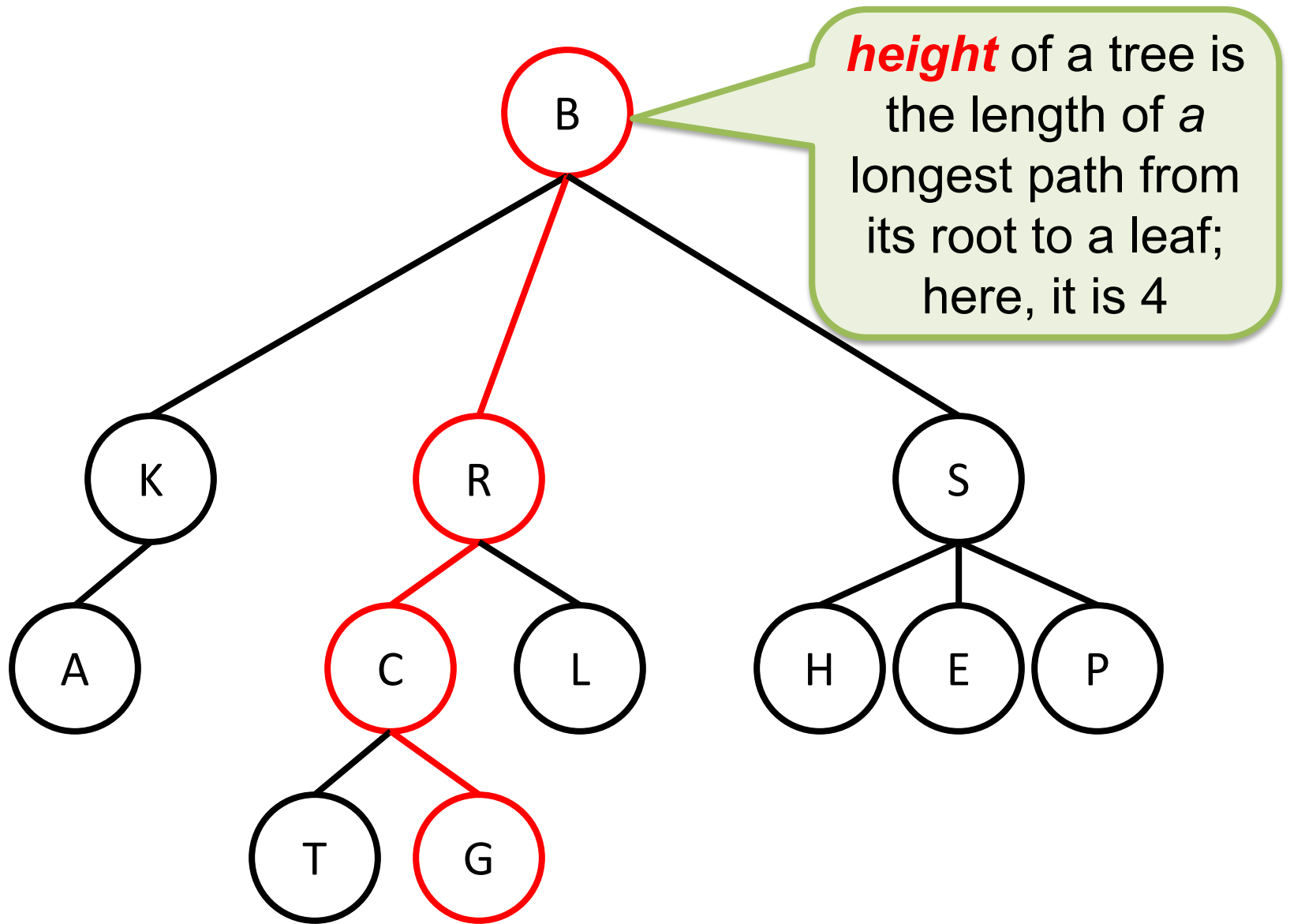


Your Turn!

What's the **size** of each of these 4 trees?

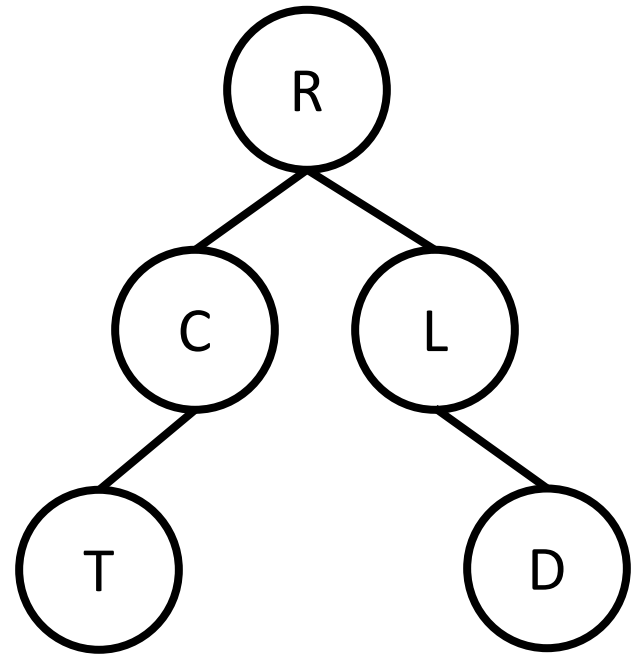
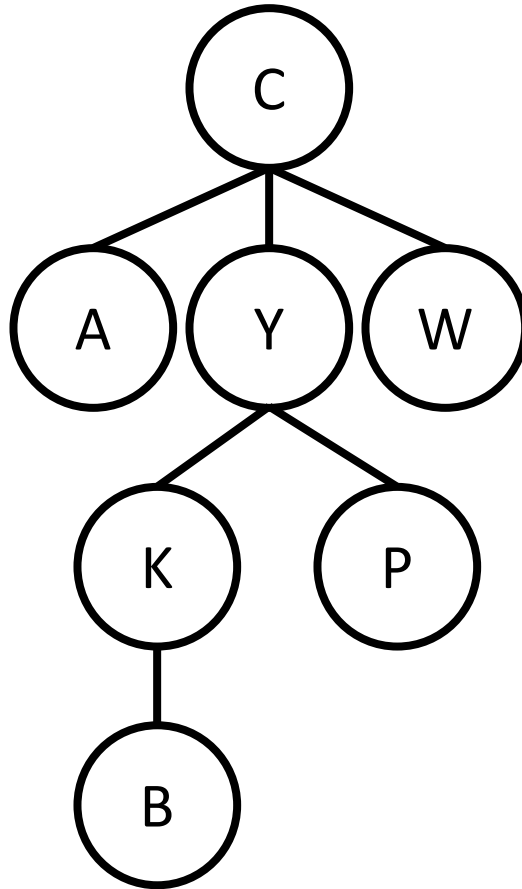
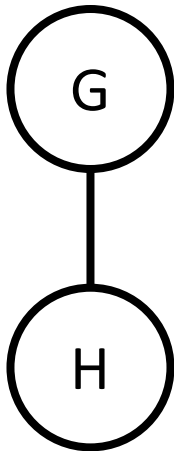
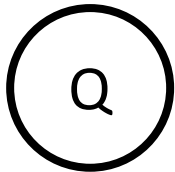






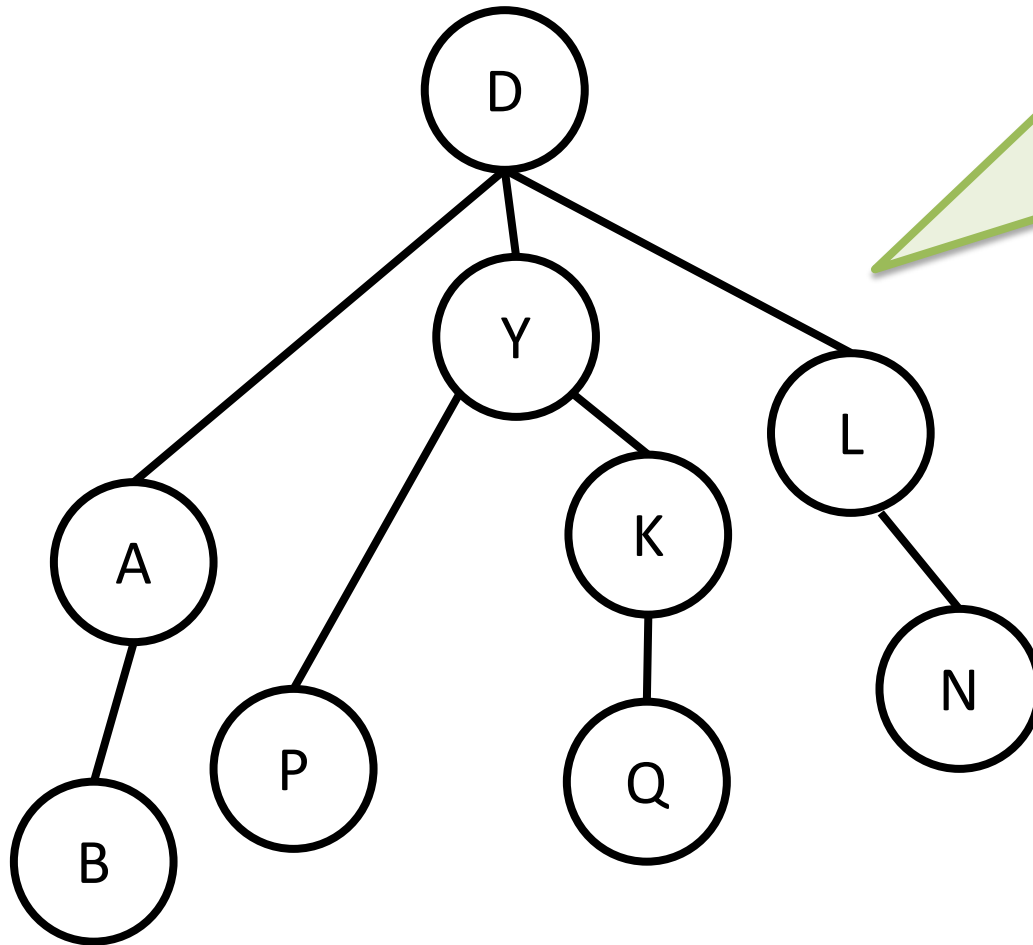
Your Turn!

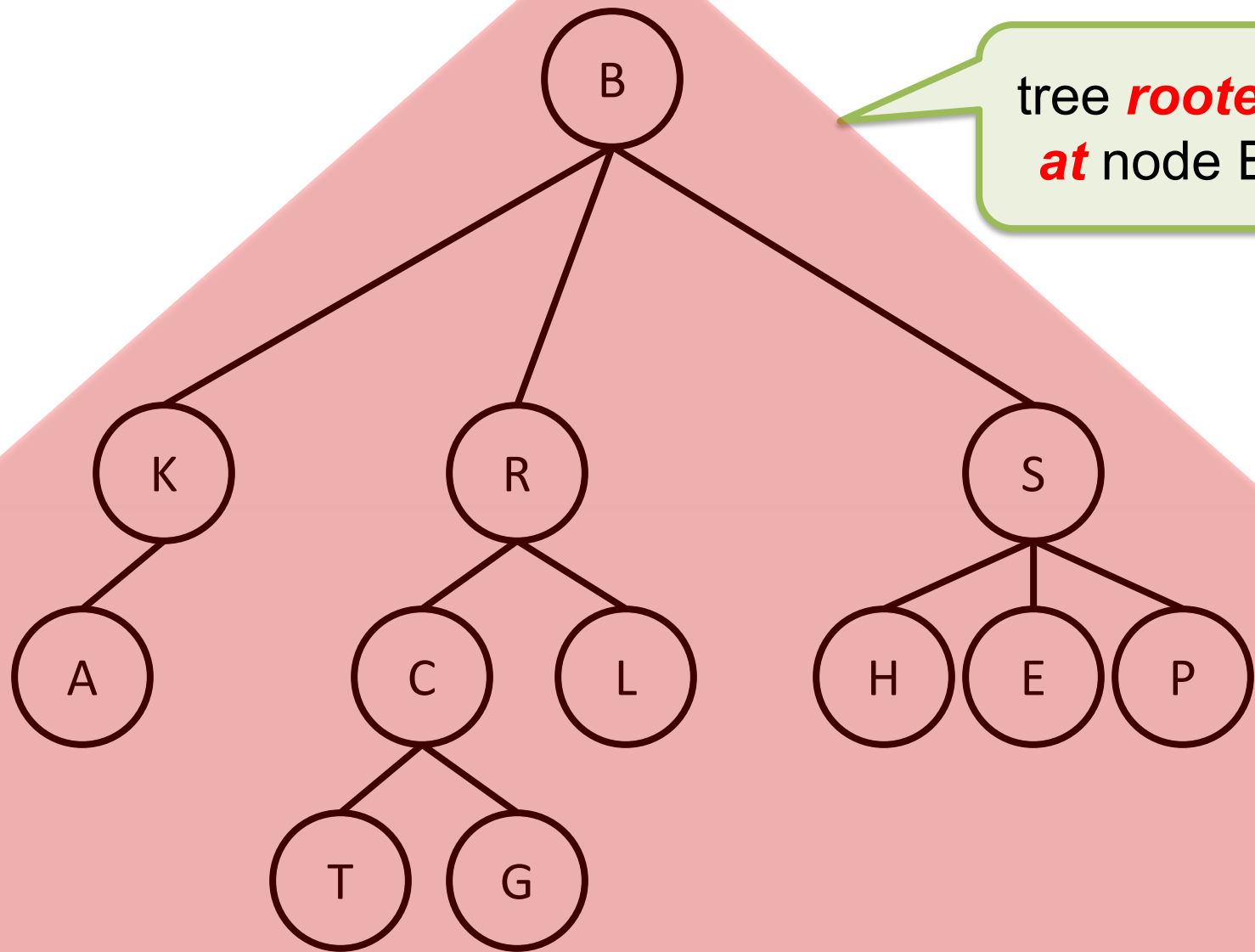
What's the *height* of each of these 4 trees?



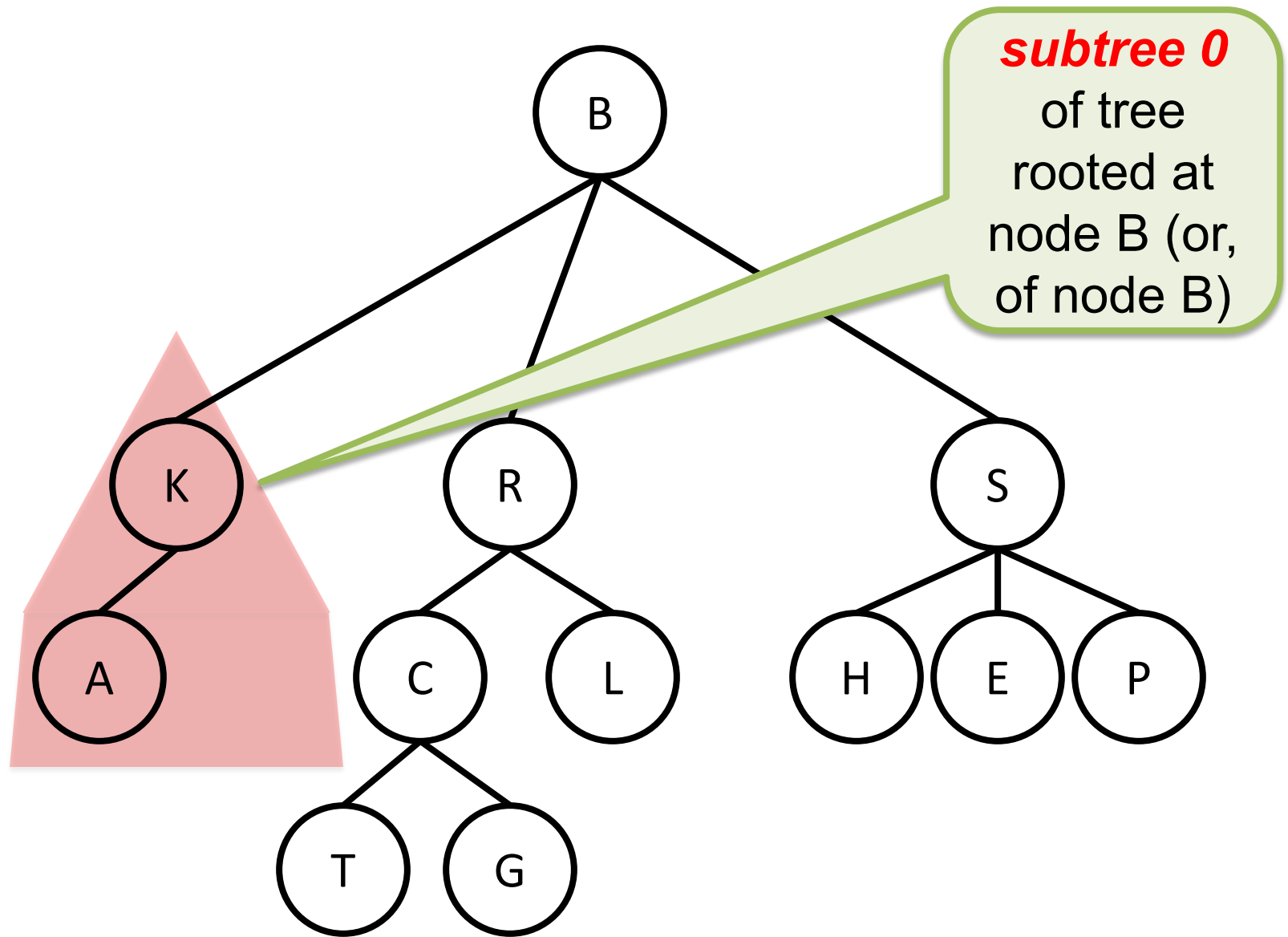
A Tricky One?

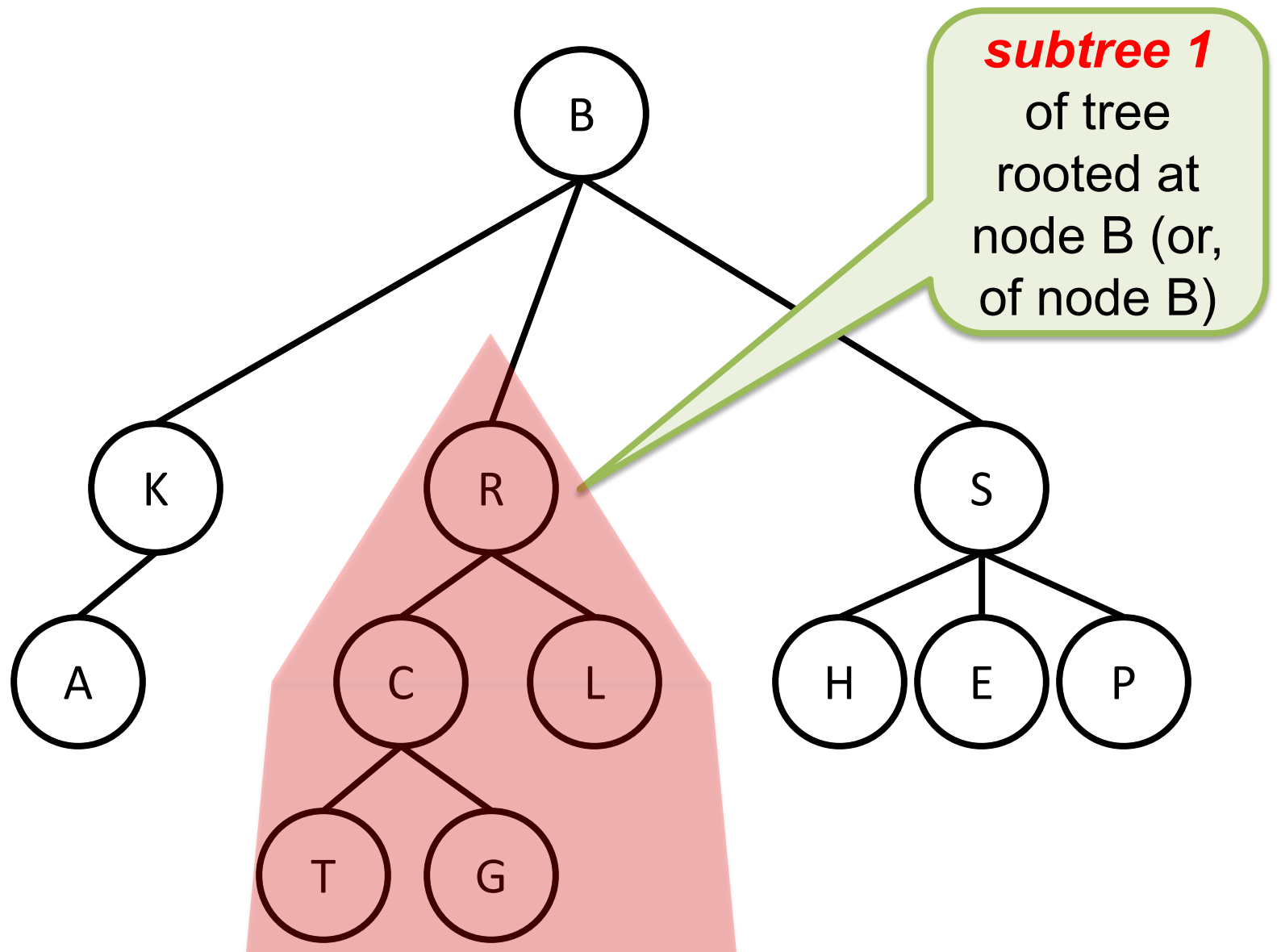
What's the **height** of this tree?

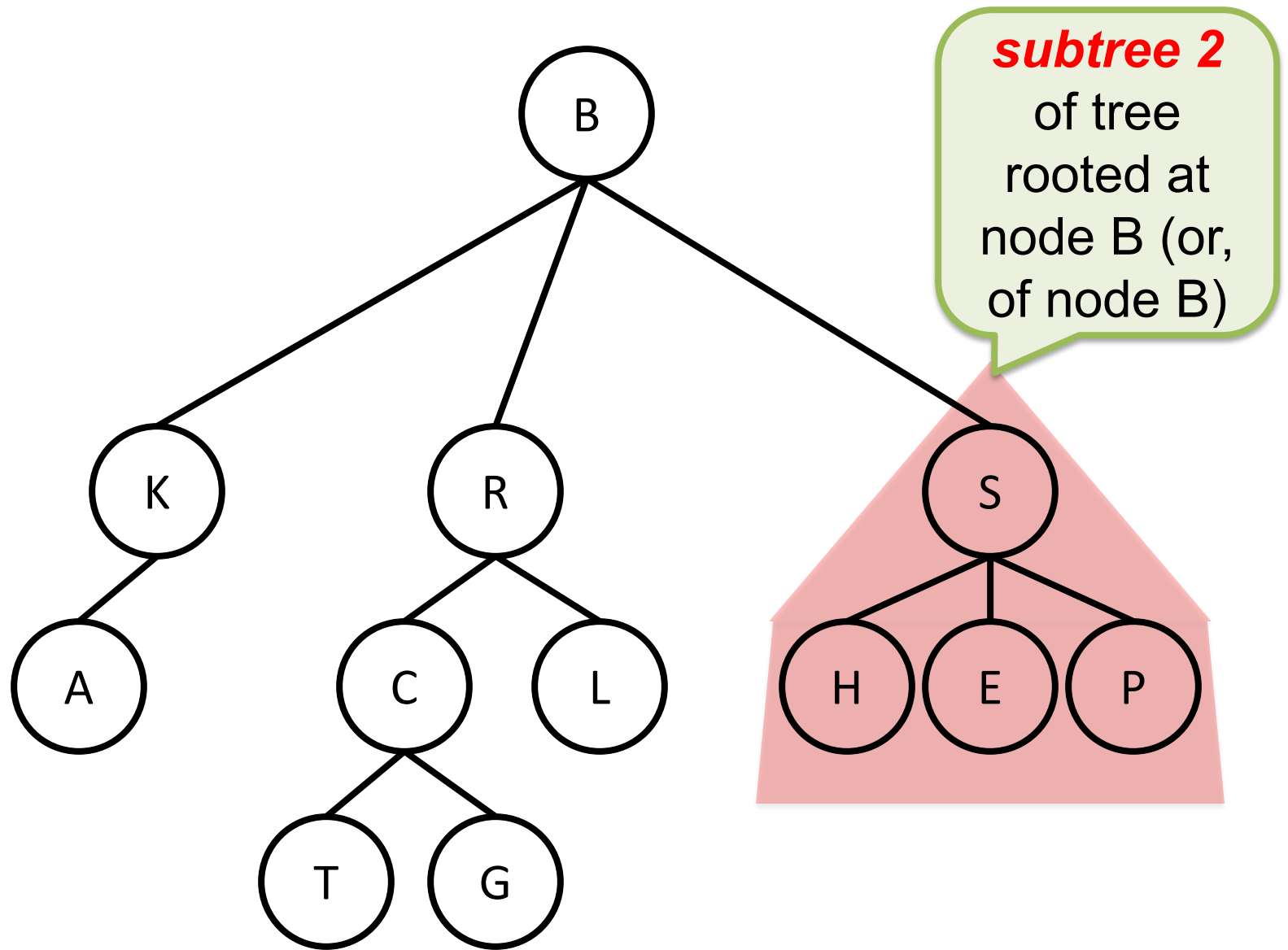


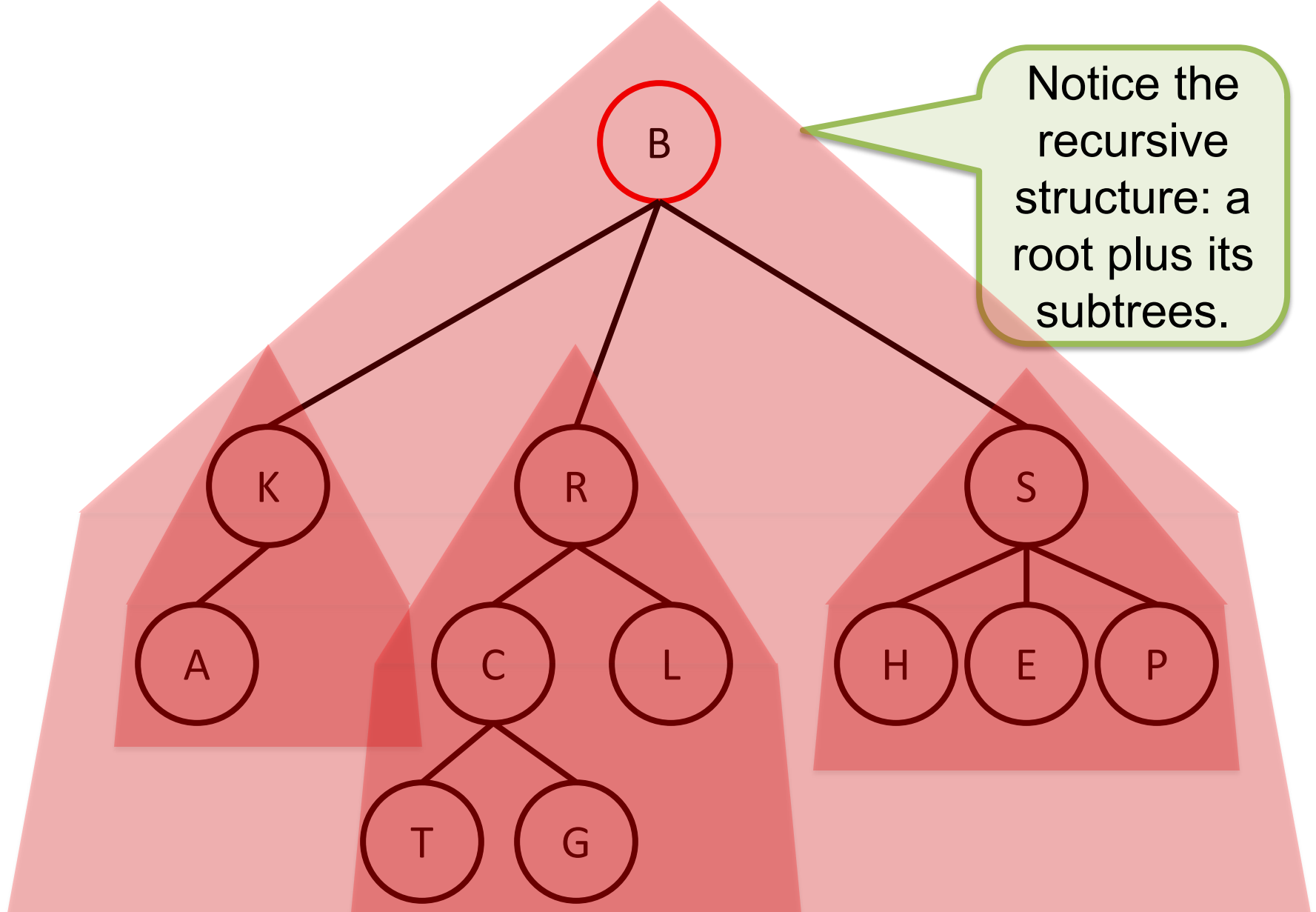


tree **rooted**
at node B









Resources

- Wikipedia: Tree structure
 - http://en.wikipedia.org/wiki/Tree_structure