Make a tree in the class ***LRCMap*** (**25%**)
The tree node can have *references to other tree nodes* and a single data variable, ***nothing else***. The tree node can ***not*** hold the key value itself. The key must be represented only by the placement of the node in the tree. *There can be data in any node, not only leaves*.
The keys are strings which only have three possible characters, but as many of those as needed. The characters are **'l'** (*left*), **'r'** (*right*) and **'c'** (*center*), so an example is: “llrcrclcclr”.
The tree class has two operations:

- put_data(key, data)
  - Places this ***data*** in the tree corresponding to this ***key***
  - Overwrite/update if data is already there.
- get_data(key)
  - returns ***data*** for that ***key***
  - returns ***None*** if non-existant
  - Returns as soon as it is evident the key is not there
    - Does not initialize any new nodes while searching

The constructor takes a boolean parameter.

- __init__(self, build = False)
  - If ***build*** is set to True, build a tree that can be used for strings of length up to 8 characters, without adding new nodes after the initialization.
  - If ***build*** is set to false, initialize an empty tree, or root only.

Example input and tree: