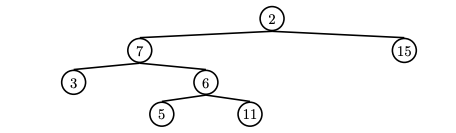
Practice 06 - Trees

**Practice 1** - Define the tree below using the tree constructor.

|  |  |
| --- | --- |

**Practice 2** - Write a function that returns the largest number in a tree.

| def tree\_max(t):  """Return the maximum label in a tree.  >>> t = tree(4, [tree(2, [tree(1)]), tree(10)])  >>> tree\_max(t)  10  """ |
| --- |

**Practice 3** - Write a function that takes in a tree and a value x and returns a list containing the nodes along the path required to get from the root of the tree to a node containing x. 

If x is not present in the tree, return None. Assume that the entries of the tree are unique.

For the following tree, find path(t, 5) should return [2, 7, 6, 5]

| def find\_path(tree, x):  """ >>> t = tree(2, [tree(7, [tree(3), tree(6, [tree(5), tree(11)])] ), tree(15)])  >>> find\_path(t, 5)  [2, 7, 6, 5]  >>> find\_path(t, 10) # returns None  """ | |
| --- | --- |
| if \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_:  return \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_:  path = \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  if \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_:  return \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | """You may also user your own structure""" |