1. Link: https://docs.google.com/document/d/1QxAPGbx\_OndBg5GGZe3t4cd6ezWtKB4WhLl5HGjUZOg/edit

def num\_nodes(t):

if is\_leaf(t):

return 1

return  1  + sum( [num\_nodes(b) for b in branches(t)] )

2. Link: http://alvinwan.com/cs61a/Quiz5102.pdf

def helper(t):

c h i l d r e n = **b r a n c h e s ( t )**

**print(root(t))**

[ h e l p e r ( b ) f o r **b** i n **c h i l d r e n** ]

helper(t)

3.

Link: https://inst.eecs.berkeley.edu/~cs61a/sp18/assets/pdfs/61a-fa17-mt2-review.pdf

**def** is\_min\_heap(t):

**for** b **in** branches(t):

**if** label(t) > label(b) **or not** is\_min\_heap(b):

**return** False

**return** True

HOF: https://cs61a.org/assets/pdfs/61a-fa17-mt1-review-sol.pdf

Return a higher-order function that prints its memories.

**def** g(h):

**print**(f(x))

**return** memory(x, h) **return** g