Quiz 3

Brian Hou

September 24, 2015

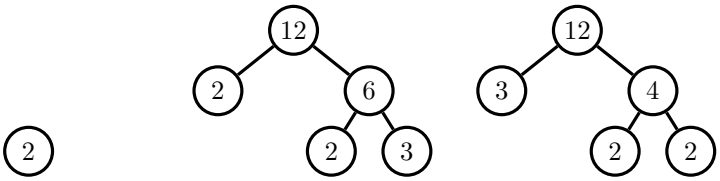
1. Draw the environment diagram.

```
def reverse(lst):
    if len(lst) <= 1:
        return lst
    return reverse(lst[1:]) + [lst[0]]

l = [1, [2, 3], 4]
rev = reverse(l)
```

2. We can represent the factorization of a number with a *full binary tree*, a tree that has either two subtrees or none at all. Implement `make_factor_tree`, which takes in an integer `n` that is greater than one and returns a tree that factors `n`.

   Example factor trees for 2 and 12 are shown below. The product of all leaves of a factor tree must be `n`. There may be multiple valid factor trees.




```
def make_factor_tree(n):
    """
    >>> six = make_factor_tree(6)
    >>> print(root(six), len(branches(six)))
    6 2
    >>> print(root(branches(six)[0]), root(branches(six)[1]))
    2 3
    >>> two = make_factor_tree(2)
    >>> print(root(two), is_leaf(two))
    2 True
    """
```