

Homework 7

CSC 121-2
Fall 2019

1. Using recursion and the function `add`, define a multiplication function

```
mult :: Nat -> Nat -> Nat
```

for natural numbers. Then, use it to define an exponentiation function

```
exp :: Nat -> Nat -> Nat.
```

2. Consider the `List` developed on page 97 (page 103 of first edition). Write a function

```
append :: List a -> List a -> List a
```

that appends two `List a` values.

3. Write a function `reverse :: List a -> List a` that reverses a `List a` value.

4. Consider the binary search tree developed on pages 97–98 (pages 103–104 of first edition). The height of the tree is the maximum distance the root from a leaf. Write a recursive function `height :: Tree -> Int` to compute the height of a binary search tree.

5. A binary tree is full if every node has exactly 0 or 2 children. Write a function

```
full :: Tree -> Bool
```

to determine if a tree is full or not.

6. A binary tree is balanced if at every node, the height of the left and right subtrees differ by at most 1. Write a function `balanced :: Tree -> Bool` to determine if a tree is balanced or not.