As always, sit with a partner and work through these together.

**Activity 1.** Circle the variables being mutated in the following function.

```python
def find_max(L):
    max = -infinity
    for i from 0 to len(L):
        if L[i] > max:
            max = L[i]
    return max
```

**Activity 2.** Solve the following reduce function call, showing each recursive step and marking the accumulator at each step.

```python
multiply = lambda x,y: x*y
reduce(multiply, [1,2,3,4,5], 1)
```
1. Complete this anonymous function that raises a single argument $n$ to the $n^{th}$ power
   
   \[
   \lambda n: \_{
   \]

2. Write a line of code that applies the function you wrote in part 1 to every element of an input list, list
   
   \[
   list
   \]

3. Complete this anonymous function that takes in a single argument $n$ and returns a function that takes in no arguments and returns $n$
   
   \[
   \lambda n: \_{
   \]

4. Write a line of code that applies the function you wrote in part 3 to an input list. This should give you a list of functions. Write another line of code that takes in the list of functions produced by your first line and turns it back into the original list.
   
   \[
   1 \text{ function\_list } = \_{
   \]
   
   \[
   2 \_{
   \]

5. Remove odd numbers from a list using reduce.
   
   \[
   \text{def remove\_odds(my\_list):}
   \]
   
   \[
   \text{ return } \_{
   \]