CS 421 --- State Activity

<table>
<thead>
<tr>
<th>Role</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manager</td>
<td>Keeps team on track</td>
</tr>
<tr>
<td>Recorder</td>
<td>Records decisions / QC</td>
</tr>
<tr>
<td>Reporter</td>
<td>Reports to class</td>
</tr>
<tr>
<td>Reflector</td>
<td>Assesses team performance</td>
</tr>
</tbody>
</table>

Please write your name/netid legibly in dark ink. Hand in one copy per team. Do not staple or mangle the corners.

Challenge

Example 1) Can you write a counter function (or function producer) without using objects and without using global variables?

Two hints: python supports HOFs and nested function declarations. To access an outer scoped variable, use the `nonlocal` keyword. Here is a convoluted `increment` function that shows some of the features you will need.

```python
def inc(x):
    i = x
    def doit():
        nonlocal i
        i = i + 1
    doit()
    return i
```

Problem 1) Consider this Python code.

```python
class Delay:
    def __init__(self, action):
        self.action = action
        self.status = 0

    def report(x):
        print("Thunk executed: {}\n".format(x))
        return x

    def force(self):
        if self.status == 2:
            return self.value
        elif self.status == 0:
            self.status = 1
            self.value = Delay.report(self.action())
            self.status = 2
            return self.value
        else:
            return Exception("It broke!"
```
• Review this code with a partner. Can you figure out how to use it?
• What is the purpose of `self.status = 1`?
Consider this function and list definition.

```python
def lazyTake(n, x):
    if x == () or n < 1:
        return ()
    else:
        return (x[0], lazyTake(n - 1, x[1].force()))
```

```python
l1 = (2, Delay(lambda: (3, Delay(lambda: (5, Delay(lambda: ())))))))
```

**Problem 2)** Can you write `lazyTail`, `lazyMap`, and `lazyZipWith`?

**Problem 3)** Use these functions to make the infinite list of natural numbers and Fibonacci numbers.