Discussion 6: Testing & Algorithmic Complexity

Testing

1. We try to test our code, but we get an error. What does it mean and how can we fix it?
Algorithmic Complexity: Definitions

1. What is runtime? How do we measure it?

2. If a function runs in $O(n)$ time, that means it runs…
   - $O$ in linear time at worst
   - $O$ in linear time on average
   - $O$ in linear time at best

Understanding Runtimes

1. Fill in the following chart:

<table>
<thead>
<tr>
<th>Runtime</th>
<th>Notation</th>
<th>As input size increases by…</th>
<th>The number of steps change by…</th>
</tr>
</thead>
<tbody>
<tr>
<td>Constant</td>
<td>x2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Logarithmic (base 2)</td>
<td>x2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Linear</td>
<td>x2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Quadratic</td>
<td>x2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Exponential (base 2)</td>
<td>+1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2. In the following diagram, which is the best runtime? The worst?
Runtime Practice

1. Find the runtime of each of the following blocks or processes.

   a. 
   
   ```
   +add+ x + and + y +
   report (x + y)
   ```

   b. 
   
   ```
   +average+ list +
   script variables sum
   for each item of list
   change sum by item
   report sum / length of list
   ```

   d. This process takes in a value and a list and searches through every item in the list one by one to see if it can find that value.

   e. This process takes in a value and a sorted list and searches for the value in the sorted list. Every iteration of the algorithm, it figures out which half of the list the value would be in, and then only searches in that half of the list.

   f. 
   
   ```
   + sort+ list +
   script variables sorted list min min_index
   set sorted list to list
   repeat until length of list = 0
   set min_index to 0
   set min to item 1 of list
   for i = 1 to length of list
   if item i of list < min
   set min to item i of list
   set min_index to i
   add item min_index of list to sorted list
   delete min_index of list
   report sorted list
   ```

   g. You know a secret, and you want to share it with the world. In state 0, you are the only person who knows the secret. Then in state 1, you share the secret with two friends, so three total people know the secret. Then in state 2, both of your friends tell two of their friends, so seven total people know the secret. This pattern (of people sharing the secret with two friends) continues indefinitely. As a function of the state, what is the order of growth of the number of people who know the secret?
What is the runtime of this block when $n$ is less than 7?

What is the runtime of the block when $n$ is greater than 7?

Why?
What do the following calls report? The first one is done for you.

```
my_func 5 5 5 5
my_func 10 10 10 10
```
Challenge Problems

1. What does the following expression do? Assuming that all helper (non-HOF) blocks operate in constant time, what is its runtime?

2. Assume that the word → list block executes in linear time as a function of the length of the input word. If myList is a list of n words, each of length n, what is the runtime of the following expression?