Recursion

Asymptotic Time\(^1\) of Operations for Collections of\( n \) Elements

<table>
<thead>
<tr>
<th></th>
<th>Trivial ArrayList</th>
<th>Good ArrayList</th>
<th>LinkedList</th>
</tr>
</thead>
<tbody>
<tr>
<td>Insert beginning</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Insert end</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Get length</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Get ( i )th element</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1. Funny Greek symbols:
   a. \( \mathcal{O} \): Upper bound: the running time is no \emph{slower} than this. To be useful, make this as small as possible. We can eat breakfast before lecture because breakfast couldn’t possibly take more than half an hour.
   b. \( \Omega \): Lower bound: the running time is no \emph{faster} than this. To be useful, make this as large as possible. We can have a party because mom won’t be home for at least 3 days.
   c. \( \Theta \): Exact bound. \( f = \Theta(g) \) means that \( f = \mathcal{O}(g) \) and \( f = \Omega(g) \).

2. More funny symbols:
   a. \( a \cdot b = \text{“}a\text{ added to itself }b\text{ times”} \)
   b. \( a^b = \text{“}a\text{ multiplied by itself }b\text{ times”} \)
   c. \( a / c = \text{“}how many times you have to add }a\text{ to itself to produce }c\text{”} \)
   d. \( \log_a c = \text{“}how many times you have to multiply }a\text{ by itself to produce }c\text{”} \)

3. Analyze:
   a. \( \text{int } f(\text{int } x) \{ \text{ if } (x <= 0) \{ \text{ return 1; } \} \text{ else } \{ \text{ return } x \cdot f(x - 1); \} \}
   b. \( \text{int } g(\text{int } x) \{ \text{ if } (x <= 0) \{ \text{ return 1; } \} \text{ else } \{ \text{ return } x \cdot g(x / 2); \} \}

---

\(^1\) Expected, amortized time