

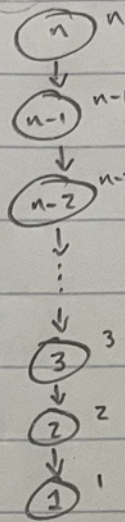
Question 3

```

1 def sum_list1(lst):
2     if (len(lst) == 1):
3         return sum
4     else:
5         rest = sum_list1(lst[1:])
6         sum = lst[0] + rest
7         return sum

```

→ $\Theta(1)$



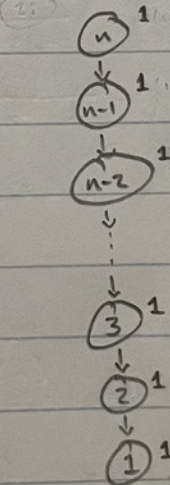
$$\begin{aligned}
 T(n) &= n + (n-1) + (n-2) + \dots + 3 + 2 + 1 \\
 &= 1 + 2 + 3 + \dots + n \\
 &= \frac{n(n+1)}{2} \\
 &= \frac{n^2(n+1)}{2} + \frac{n}{2} \\
 &= \frac{n^3}{2} + \frac{n^2}{2} + \frac{n}{2} \\
 &= \Theta(n^2)
 \end{aligned}$$

```

1 def sum_list2(lst, low, high):
2     if (low == high):
3         return lst[low]
4     else:
5         rest = sum_list2(lst, low+1, high)
6         sum = lst[low] + rest
7         return sum

```

→ $\Theta(1)$



$$\begin{aligned}
 T(n) &= \underbrace{1 + 1 + \dots + 1 + 1 + 1}_{n \text{ times}} \\
 &= \Theta(n)
 \end{aligned}$$

3.) Asymptotically, version #2 (sum_list2) is faster.