Recursion

Asymptotic Time[1] of Operations for Collections of *n* Elements

	Trivial ArrayList	Good ArrayList	LinkedList
Insert beginning			
Insert end			
Get length			
Get i^{th} element			

1. Funny Greek symbols:
   a. O: Upper bound: the running time is no *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. Ω: Lower bound: the running time is no *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. Θ: Exact bound. $f = \Theta(g)$ means that $f = O(g)$ and $f = \Omega(g)$.

2. More funny symbols:
   a. $a * b$ = "*a* **added** to itself *b* times"
   b. a^b = "*a* **multiplied** by itself *b* times"
   c. a / c = "how many times you have to **add** *a* to itself to produce *c*"
   d. $\log_a c$ = "how many times you have to **multiply** *a* by itself to produce *c*"

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

[1] Expected, amortized time