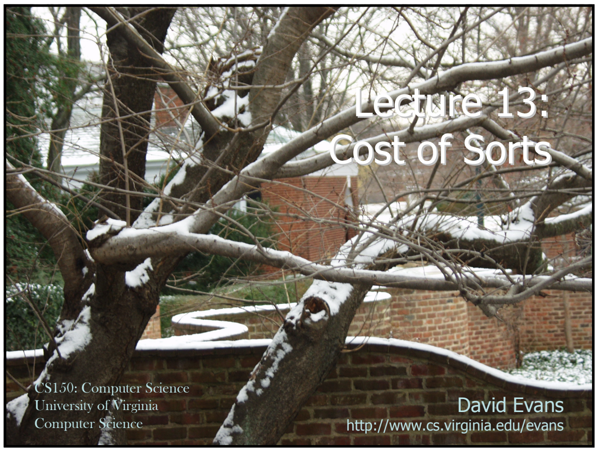

Menu

- PS3, Question 4
- Operators for Measuring Growth
- Sorting

Sorting Cost

```
(define (sort lst cf)
  (if (null? lst) lst
      (let ((best (find-best lst cf)))
        (cons best (sort (delete lst best) cf)))))
(define (find-best lst cf)
   (if (= 1 (length lst)) (car lst)
       (pick-better cf (car lst) (find-best (cdr lst) cf))))
```

If we double the length of the list, the amount of work *approximately* quadruples: there are twice as many applications of find-best, and each one takes twice as long

Growth Notations

- $g \in O(f)$ ("Big-Oh")
  - g grows **no faster than** f (upper bound)
- $g \in \Theta(f)$ ("Theta")
  - g grows **as fast as** f (tight bound)
- $g \in \Omega(f)$ ("Omega")
  - g grows **no slower than** f (lower bound)

Which one would we most like to know?

Meaning of O ("big Oh")

g is in $O(f)$ iff:

There are positive constants c and n_0 such that

$$g(n) \leq cf(n)$$

for all $n \geq n_0$.

O Examples

g is in $O(f)$ iff there are positive constants c and n_0 such that $g(n) \leq cf(n)$ for all $n \geq n_0$.

Is n in $O(n^2)$? Yes, $c = 1$ and n_0=1 works.

Is $10n$ in $O(n)$? Yes, $c = .09$ and n_0=1 works.

Is n^2 in $O(n)$? No, no matter what c we pick, $cn^2 > n$ for big enough n ($n > c$)

Ω ("Omega"): Lower Bound

> g is in $O(f)$ iff there are positive constants c and n_0 such that $g(n) \le cf(n)$ for all $n \ge n_0$.

g is in $\Omega(f)$ iff there are positive constants c and n_0 such that

$$g(n) \geq cf(n)$$

for all $n \ge n_0$.

Charge

- Read Chapter 6 and 7 of the course book
- PS4 is due Monday