bounded below

Notice that the definition of big-Omega differs in just one character from big-Oh:
\[ \Omega(g) = \{ f : \mathbb{N} \rightarrow \mathbb{R}_{\geq 0} \mid \exists c \in \mathbb{R}^+, \exists B \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B \Rightarrow f(n) \geq cg(n) \} \]

The rôle of \( B \) is, as with big-Oh, to act as a breakpoint, so comparisons don’t have to start at the origin.

The rôle of \( c \) is to scale \( g \) down below \( f \).

If you’re proving \( f \in \Omega(g) \), you get to choose \( c \) and \( B \) to suit your proof. Notice that it would be really unfair to allow \( c \) to be zero.
one last bound

It often happens that functions are bounded above \textit{and} below by the same function. In other words, \( f \in \mathcal{O}(g) \land f \in \Omega(g) \). We combine these two concepts into \( f \in \Theta(g) \).

\[
\Theta(g) = \{ f : \mathbb{N} \rightarrow \mathbb{R}^{\geq 0} \mid \exists c_1 \in \mathbb{R}^+, \exists c_2 \in \mathbb{R}^+, \exists B \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B \Rightarrow c_1 g(n) \leq f(n) \leq c_2 g(n) \}
\]

You might want to draw pictures, and conjecture about appropriate values of \( c_1, c_2, B \) for \( f = 5n^2 + 15 \) and \( g = n^2 \).
How do you deal with a general statement about two functions:

\[(f \in O(g) \land g \in O(h)) \Rightarrow f \in O(h)\]
scratch
How about: $f \in \mathcal{O}(g) \Rightarrow g \in \Omega(f)$
scratch
Prove or disprove: \( f \in \Theta(g) \Rightarrow f \cdot f \in \Theta(g \cdot g) \).

Assume \( f \in \Theta(g) \).

Then \( \exists c \in \mathbb{R}^+, \exists B \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B \Rightarrow f(n) \leq c \cdot g(n) \).

Pick \( c_1 \in \mathbb{R}^+, B_1 \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B_1 \Rightarrow f(n) \leq c_1 \cdot g(n) \).

Pick \( c = \frac{c_1^2}{c} \). Then \( c \in \mathbb{R}^+ \).

Assume \( n \in \mathbb{N} \).

Assume \( n \geq B \).

Then \( (f \cdot f)(n) = f(n) \cdot f(n) \leq f(n) \cdot c_1 \cdot g(n) \leq f(n) \cdot c_1 \cdot g(n) \leq c_1 \cdot f(n) \leq c_1 \cdot f(n) \cdot g(n) \).

\[ \leq c \cdot g(n) \cdot g(n) \leq c \cdot g(n) \cdot g(n) \leq c \cdot g(n) \cdot g(n) = c(g \cdot g)(n) \leq c(g \cdot g)(n) \]

Then \( \exists c \in \mathbb{R}^+, \exists B \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B \Rightarrow (f \cdot f)(n) \leq c(g \cdot g)(n) \).

Then \( f \cdot f \in \Theta(g \cdot g) \).

Thus, \( f \cdot f \in \Theta(g \cdot g) \).

\text{slide 8}
scratch
Prove or disprove: \((f \in O(h) \land g \in O(h)) \implies (f + g) \in O(h)\).

Assume \(f \in O(h) \land g \in O(h)\) \# antecedent.

Then \(\exists c \in \mathbb{R}^+ \land \exists B \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B \implies f(n) \leq c_1 h(n)\).

Pick \(c_1 \in \mathbb{R}^+, B_1 \in \mathbb{N}\), \(\forall n \in \mathbb{N}, n \geq B_1 \implies f(n) \leq c_1 h(n)\).

Then \(\exists c \in \mathbb{R}^+ \land \exists B \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B_2 \implies g(n) \leq c_2 h(n)\).

Pick \(c_2 \in \mathbb{R}^+, B_2 \in \mathbb{N}\), \(\forall n \in \mathbb{N}, n \geq B_2 \implies g(n) \leq c_2 h(n)\).

Pick \(c = \frac{c_1 + c_2}{\max(B_1, B_2)}\) \# Then \(B \in \mathbb{N}\).

Assume \(n \in \mathbb{N}\) \# general.

Assume \(n \geq B\) \# antecedent.

Then \((f + g)(n) = f(n) + g(n) \leq c_1 h(n) + c_2 h(n) \# \text{by assumptions}\)

\leq (c_1 + c_2) h(n) \# \text{since } n \geq B_1, B_2

= (c_1 + c_2) h(n) \# c = c_1 + c_2

= c h(n) \# \text{then follow some sort of conclusion as 2 slide previous.}
scratch
Prove or disprove: \( f \in O(g) \Rightarrow f \in O(g \cdot g) \)

Pick \( f = g = \frac{1}{n+1} \). Then \( f, g \in f \cdot \frac{1}{n+1} \rightarrow (0, 1) \forall n \in \mathbb{N} \).

Then \( f \in O(g) \) is easy to prove, \( c = 1, b = 0 \) will work.

Assume \( c \in \mathbb{R}^+, b \in \mathbb{N} \) is generic, \( c \) is real, natural number.

Pick \( n = \frac{c+1}{cB} \). Then \( n \in \mathbb{N} \land n \geq B \).

Then \( f(n) = \frac{1}{n+1} \)

\[ = \frac{n+1}{(n+1)^2} \quad \text{# mult by } \frac{n+1}{n+1} \]

\[ > \frac{c+1}{(n+1)^2} \quad \# n \geq c \]

\[ > \frac{c}{(n+1)^2} \quad \# \frac{c+1}{c} \geq c, \quad \# \text{so } \frac{c+1}{c} \geq c \]

\[ = c \cdot (g \cdot g)(n) \]

Then \( \exists n \in \mathbb{N}, n \geq B \land f(n) > c(g \cdot g)(n) \) \# introduced \( \exists \)

Then \( \forall c \in \mathbb{R}^+, \forall B \in \mathbb{N}, \exists n \in \mathbb{N}, n \geq B \land f(n) > c(g \cdot g)(n) \) \# introductory axiom.

Then \( f \not\in O(g \cdot g) \) \# violates definition.

Then \( \exists f, g \in f, f \in O(g) \land f \not\in O(g \cdot g) \).
scratch
counting costs

want a coarse comparison of algorithms “speed” that ignores hardware, programmer virtuosity

which speed do we care about: best, worst, average? why?

define idealized “step” that doesn’t depend on particular hardware and idealized “time” that counts the number of steps for a given input.
linear search

```python
def LS(A, x):
    """ Return index i such that x == L[i]. Otherwise, return -1 """
    i = 0
    while i < len(A):
        if A[i] == x:
            return i
        i = i + 1
    return -1
```

Trace LS([2, 4, 6, 8], 4), and count the time complexity $t_{LS}([2, 4, 6, 8], 4) = 7$ if $3j + 3/2 < 4$

What is $t_{LS}(A, x)$, if the first index where $x$ is found is $j$?

What is $t_{LS}(A, x)$ is $x$ isn’t in $A$ at all?

slide 15
worst case

denote the worst-case complexity for program $P$ with input $x \in I$, where the input size of $x$ is $n$ as

$$W_P(n) = \max \{ t_P(x) \mid x \in I \land \text{size}(x) = n \}$$

The upper bound $W_P \in \Theta(U)$ means

$$\exists c \in \mathbb{R}^+, \exists B \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B \Rightarrow \max \{ t_P(x) \mid x \in I \land \text{size}(x) = n \} \leq cU(n)$$

That is:

$$\exists c \in \mathbb{R}^+, \exists B \in \mathbb{N}, \forall x \in I, \text{size}(x) \geq B \Rightarrow t_P(x) \leq cU(\text{size}(x))$$

The lower bound $W_P \in \Omega(L)$ means

$$\exists c \in \mathbb{R}^+, \exists B \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B \Rightarrow \max \{ t_P(x) \mid x \in I \land \text{size}(x) = n \} \geq c\Omega(n)$$

That is:

$$\exists c \in \mathbb{R}^+, \exists B \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B \Rightarrow \exists x \in I, \text{size}(x) = n \land t_P(x) \geq cL(n)$$
def IS(A):
    """IS(A) sorts the elements of A in non-decreasing order """
    i = 1
    while i < len(A):
        t = A[i]
        j = i
        while j > 0 and A[j-1] > t:
            j = j-1
        A[j] = t
        i = i+1

I want to prove that $W_{IS} \in \mathcal{O}(n^2)$.

$\exists c \in \mathcal{R}^1, \exists B \in \mathbb{N}, \forall x \in I, \; \text{size}(x) \geq B \Rightarrow t_{IS}(x) \leq c \cdot n^2$
Pick \( c = 1 \) Then \( c \in \mathbb{R}^+ \)

Pick \( B = 1 \) Then \( B \in \mathbb{N} \).

Assume \( x \) is an array and \( \text{len}(x) = n \geq B \).

For Lines 5, 6, 7 execute once as \( i \) decrements
from \( 1 \) to \( ? \) \( 1 \) \((+1 \text{ loop condition})\)

yield \( 3i+1 \) steps \( \leq 3n+1 \) steps

for each \( i \).

Then \( i \) takes values \( 1, \ldots, n-1 \), yielding
an addition each of \( 2, 3, 4, 8, 9 \) each
times, hence \( (n-1)(5+(3n+1)) + 1 + 1 \)

\( \leq n (4 + 3n) + 1 = 3 n^2 + 6n + 2 \)

\( \leq 11n^2 \quad \# \quad n \geq 1, n \geq 1 \)

\( = cn^2 \quad \# \quad 11 = c \)
scratch
$h(n+1) \over 2$

$1 + \cdots + n - 1$

$\chi = [n-1, n-2, \ldots, 0]$

$1 + \cdots + n$

scratch
scratch