We can also show that functions are not big-O.

Example: n^2 is not $O(n)$.

Proof: By contradiction. Suppose that n^2 is $O(n)$. Then there exist constants c and n_0 such that $n^2 \leq c \cdot n$ for all $n \geq n_0$.

In particular, $n_0^2 \leq c \cdot n_0 \Rightarrow n_0 \leq c$.

Take $n = c+1$. Since $n_0 \leq c < c+1$, we should have that $(c+1)^2 \leq c \cdot (c+1)$. But then $c+1 \leq c$, which is impossible. Therefore n^2 cannot be $O(n)$.

Big-O in practice

Problem (Element distinctness)

Input: Sequence $A = a_1, a_2, \ldots, a_n$.

Output: YES if there are i, j such that $a_i = a_j$.
NO otherwise.

Algorithm 1

```
1. for i=1 to n do                          ≤ n×
2.   for j=i+1 to n do                      ≤ n×  (Σ_{j=i+1} O(1))
3.     if a_i = a_j then                    } O(1)
4.       return YES                         }
5. return NO                                O(1)
```

Total: $T_1(n) \leq n \times n \times O(1) + O(1)$
$= n^2 \times O(1) + O(1)$
$= O(n^2) + O(1)$
$= O(n^2 + 1)$
$= O(n^2)$

Algorithm 2

```
1. Sort A                                   O(n log n)
2. for i=1 to n-1 do                        ≤ n×
3.   if a_i = a_{i+1} then                  } O(1)
4.     return YES                           }
5. return NO                                O(1)
```

Total: $T_2(n) \leq O(n \log n) + n \times O(1) + O(1)$
$= O(n \log n) + O(n) + O(1)$
$= O(n \log n + n + 1)$
$= O(n \log n)$

Rules for big-O

Assume all functions are positive ($\forall n \geq 0 (f(n) > 0)$)

1. If $f(n)$ is $O(g(n))$ then $k \times f(n)$ is $O(g(n))$ for $k \in \mathbb{R}^+$.

Proof: $k \times f(n) \leq k \times c' \times g(n)$ (for $n \geq n_0'$)

$= (k \times c') \times g(n)$

Therefore $k \times f(n)$ is $O(g(n))$ with $c = k \times c'$ and $n_0 = n_0'$.

2. If $f_1(n)$ is $O(g_1(n))$ and $f_2(n)$ is $O(g_2(n))$ then $f_1(n) \times f_2(n)$ is $O(g_1(n) \times g_2(n))$.

Proof: $f_1(n) \times f_2(n) \leq c_1 \times g_1(n) \times f_2(n)$ (for $n \geq n_1$)

$\leq c_1 \times g_1(n) \times c_2 \times g_2(n)$ (for $n \geq n_2$)

$= (c_1 \times c_2) \times (g_1(n) \times g_2(n))$

Therefore $f_1(n) \times f_2(n)$ is $O(g_1(n) \times g_2(n))$ with $c = c_1 \times c_2$ and $n_0 = \max(n_1, n_2)$.

3. If $f_1(n)$ is $O(g_1(n))$ and $f_2(n)$ is $O(g_2(n))$ then $f_1(n) + f_2(n)$ is $O(g_1(n) + g_2(n))$.

Proof: In tutorial.

4. If $f(n)$ is $O(g(n))$ and $g(n)$ is $O(h(n))$ then $f(n)$ is $O(h(n))$.

Proof: $f(n) \leq c_1 \times g(n)$ (for $n \geq n_1$)

$\leq c_1 \times c_2 \times h(n)$ (for $n \geq n_2$)

Therefore $f(n)$ is $O(h(n))$ with $c = c_1 \times c_2$ and $n_0 = \max(n_1, n_2)$

Consequence: If $f(n)$ is $O(g(n))$ and $g(n)$ is $O(h(n))$ then $f(n) + g(n)$ is $O(h(n))$.

Proof: ~~$f(n) + g(n) \leq$ [illegible] $c_1 \times g(n) + g(n)$ (for $n \geq n_1$)~~

~~$= (c_1 + 1) \times g(n)$~~

~~$\leq (c_1 + 1) \times c_2 \times h(n)$~~

$f(n)$ is $O(h(n))$ (rule 4)

$f(n) + g(n)$ is $O(h(n) + h(n)) = O(2h(n))$ (rule 3)

$2h(n)$ is $O(h(n))$ (rule 1)

$f(n) + g(n)$ is $O(h(n))$ (rule 4)

This means that adding up "in" "big-O world" is like taking the maximum:

$O(n^2 + n) = O(n^2)$, because n is $O(n^2)$.

Function Order

- $O(1)$ = "constant"
- $O(\log n)$ "logarithmic"
- $O(\log^k n)$ $(k > 1)$; "polylogarithmic"
- $O(n^a)$ $(0 < a < 1)$
- $O(n)$ "linear"
- $O(n \log n)$
- ~~$O(n^2)$ "quadratic"~~
- $O(n^b)$ $(b > 1)$ "quadratic", etc.
- ~~$O(2^n)$ "exponential"~~
- $O(k^n)$ $(k > 1)$

(polynomial)

- $O(n!)$
- $O(n^n)$

Higher functions are big-O of lower functions.

Proof by Induction

Suppose we want to prove a proposition $P(n)$ with domain the natural numbers:

$$\forall_{n \in \mathbb{N}} (P(n))$$

As an example, let's use $P(n) = \text{“}\sum_{i=0}^{n} i = \frac{n(n+1)}{2}\text{”}$.

$\forall_{n \in \mathbb{N}} (P(n)) \equiv P(0) \wedge P(1) \wedge P(2) \wedge \ldots$

We can check some cases:

$P(0)$ is true: $\sum_{i=0}^{0} i = 0 = \frac{0 \cdot 1}{2}$ ✓

$P(1)$ is true: $\sum_{i=0}^{1} i = 0 + 1 = 1 = \frac{1 \cdot 2}{2}$ ✓

$P(2)$ is true: $\sum_{i=0}^{2} i = 3 = \frac{2 \cdot 3}{2}$ ✓

$\vdots$

$P(100)$ is true: $\sum_{i=0}^{100} i = \frac{100 \cdot 101}{2}$ ✓

Is $P(101)$ true?

$$\sum_{i=0}^{101} i = \underbrace{0 + 1 + 2 + \ldots + 100}_{\text{Of}} + 101 = \sum_{i=0}^{100} i + 101$$

$$= \frac{100 \cdot 101}{2} + 101$$

$$= \frac{100 \cdot 101}{2} + \frac{2 \cdot 101}{2}$$

$$= \frac{102 \cdot 101}{2}$$

Yes!

We can do this for every number: once we know that $P(0) \wedge \ldots \wedge P(k)$ are true, $P(k+1)$ must be true as well:

$$\sum_{i=0}^{k+1} i = \sum_{i=0}^{k} i + (k+1)$$
$$= \frac{k(k+1)}{2} + (k+1)$$
$$= \frac{k(k+1)}{2} + \frac{2(k+1)}{2}$$
$$= \frac{(k+2)(k+1)}{2}$$

$$\boxed{P(0) \wedge \ldots \wedge P(k) \Rightarrow P(k+1)}$$

But then it must be true for every number:

$P(0)$ is true (we checked this)
Therefore $P(1)$ is true.
Therefore $P(2)$ is true.
⋮
Therefore $P(n)$ is true. (for any $n \in \mathbb{N}$)

This is a proof by induction.

Structure of proofs by induction

Every proof by induction has three components:

Statement: $P(n)$ is true for all $n \geq a$.

Base case: $P(a)$ is true.

Inductive Hypothesis: Assume that $P(a) \wedge \ldots \wedge P(k)$ are all true, for some $k \in \mathbb{N}$.

Inductive Step: Prove that $P(k+1)$ is true, using the fact that $P(a), P(a+1), \ldots, P(k)$ are all true.

Example: Prove that $2^n < n!$ for all $n \geq 4$.

Proof: By induction on n.

Base case ($n=4$): $2^4 = 2\times2\times2\times2 = 16$, $4! = 1\times2\times3\times4 = 24$. $16 < 24$ ✓

Inductive Hypothesis: Assume that $2^n < n!$ for all $4 \leq n \leq k$.

Inductive Step: We need to show that $2^{k+1} < (k+1)!$.

$$\begin{aligned} 2^{k+1} &= 2 \cdot 2^k \\ &< 2 \cdot k! \quad \text{(by the inductive hypothesis)} \\ &\leq (k+1) \cdot k! \quad \text{(since } k \geq 4\text{)} \\ &= (k+1)! \end{aligned}$$

Therefore $2^n < n!$ for all $n \geq 4$.

(This shows that 2^n is $O(n!)$).

Exercise: Prove that for all integers $n \geq 1$, $n^3 - n$ is divisible by 3.

Proof: By induction on n.

Base case ($n=1$): $1^3 - 1 = 0 = 0 \times 3$ is divisible by 3.

Inductive Hypothesis: Assume that $n^3 - n$ is divisible by 3 for all $1 \leq n \leq k$.

Inductive Step: We need to show that $(k+1)^3 - (k+1)$ is divisible by 3.

$$\begin{aligned} (k+1)^3 - (k+1) &= (k^3 + 3k^2 + 3k + 1) - (k+1) \\ &= \underbrace{(k^3 - k)}_{\text{IH: div. by 3}} + \underbrace{3(k^2 + k)}_{\text{div. by 3}} \end{aligned}$$

Therefore $n^3 - n$ is div. by 3 for all $n \geq 1$.

end of Lec