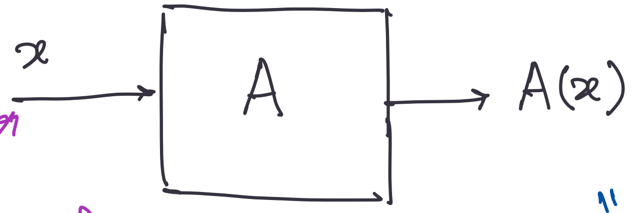
Algorithm Analysis and Design. (Monsoon 2022)

→ "Non-interactive system".



- Representation of input ← Structured input
- "Less time and space complexity".
- Ex: Array, Adjacency list

→ Bubblesort: $O(n^2)$ ~~time~~ and $O(n)$ space.

↑ # of comparisons

n ← # of elems of the array.

→ Matrix multiplication: A, B of order $n \times n$.

$C = A \cdot B$ $\qquad$ $O(n^3)$

↑ operations.

$$i,j \in [1,n] : \underline{\underline{C_{ij}}} = \sum_{k=1}^{n} \underline{A_{ik} \cdot B_{kj}}$$

n multiplications

n additions

Karatsuba's integer mult. [2 ℓ-bit numbers, complexity of mult $\sim O(\ell \log \ell)$.

↳ $n^2 \cdot n \cdot [O(\ell \log \ell) + O(\ell)]$

→ TODO: $O(?)$

Question: "When is an algorithm efficient?"

→ Optimised time and space complexity.

- ↳ "Minimum no. of operations". }
- ↳ "memory usage should be minimal".

	I_1	I_2	I_3	I_j	I_m
A_1					
A_2					
⋮					
A_i	-			*	
A_s					

$$\frac{\max_j \{A_i(I_j)\}}{\operatorname{avg}_j \{A_i(I_j)\}}$$

* ← "running time" of Algorithm A_i on instance I_j.

→ Quicksort: want to Choose a pivot that breaks array into two "halves".

→ First element

"Worst-case running time".

"Real world instances seem to have better performances and worst case instances are hardly encountered".

- → Linear programming
- → Satisfiability (Given a Boolean formula, we seek a satisfiable assignment)

"Average-case analysis".

→ "Probability distributions over instances". } "Tricky"

↑ Apriori not clear on how to define this

Problem: "No efficient way to define prob".

or

"Cannot gauge if equal prob. is the right notion".

"Worst-case analysis".

→ "An algorithm is efficient if it achieves qualitatively better performance than brute-force approach" in the worst case.

→ Qu: Can you give an algo to count the no of triangles in an undir. graph?

↳ $O(n^{\omega})$ in the worst-case.

$i_1, i_2, i_3 \in V(G)$; $(i_1, i_2), (i_2, i_3)$ and $(i_3, i_1) \in E(G)$.

- For all nodes
  → go to the neighbour
  for each neighbour

$\underline{O(V^3)}$.

→ Brute-force: Run through $\binom{|V|}{3}$ many subsets of V.

→ (i_1, i_2, i_3). Check if $(i_1, i_2), (i_2, i_3)$ and $(i_3, i_1) \in E$.

matrix mult. exponent.

$\underbrace{A, A^2, A^3}$ $O(n^{\omega})$ → Strassen $\underline{\omega = \log_2 7}$

↓

[Alman Virginia-Williams] $\omega = 2.37\ldots$

$O(n^3), O(n^{\log_2 7}), \ldots, O(n^{2.3\ldots})$

Open-problem: Show that matrix mult. can be done in time $O(n^{2+\varepsilon})$ where ε is as close to 0 as possible.

"Worst-case does not always mean brute-force".

Notion of efficiency: Polynomial time.

→ We say an algorithm A is polynomial time computable if $\exists$ constants $c, d \in \mathbb{R}_{>0}$ and $n_0 \in \mathbb{N}$ s.t on every input of size $n \geq n_0$, the algorithm's running time is at most $c \cdot n^d$.

→ Sorting is in polytime. $O(\underline{n}^2)$ ~~time~~ comparisons.

A_1	$n = 100$	10000	comparisons	n^2
A_2	"	30000	——	$3n^2$
A_3	"	90000	comparisons ↘	$\frac{n^3 g}{100}$ or $n^2 \cdot g$

$\boxed{2^n} < n^{1000} \quad \forall \; \frac{n}{\log n} < 1000 \}$.

Qn: $\underline{2^n}$ or $\underline{n^{1000}}$ better? $\}$ n

$\frac{n_0}{\log n_0} \approx 1000$

n_0 s.t $\forall \; \underline{n} \geq \underline{n_0}$, $\underline{2^n \geq n^{1000}}$.

Asymptotic analysis: O, Ω, Θ

$n \quad , \quad f(n)$ ✓

$T(n) \leq f(n)$.

$\forall n \geq n_0, \; T(n) \leq f(n)$.

- $T(n) = O(f(n))$

$\exists$ a const c and $n_0 \in \mathbb{N}$ s.t $\forall \; n \geq n_0, \; T(n) \leq c \cdot f(n)$.

- If sorting is taking $O(n^2)$ then, $\exists$ a constant $c, n_0 \in \mathbb{N}$ s.t $\forall\, n \geq n_0$, sorting takes at most $c \cdot n^2$ time.

- Graph algorithms (Chapter 3 of Kleinberg-Tardos)
- Greedy algorithms (4)
- Divide and Conquer (5)
- Dynamic programming (6)
- Network flows (7)

- NP and computational intractibility
- Intro to $\left.\begin{array}{l}\text{Approx}\\ \text{Random}\\ \text{Quantum}\end{array}\right\} \rightarrow$ Advanced Algorithms.

Grading scheme:

Quiz 1 and 2	⟵	20%
Midsem		20%
End sem		30%
In-class quizzes		15%
Assignment		15%

3 textbooks

- Kleinberg Tardos
- CLRS
- Dasgupta, Papadimitrou and Vazirani