CMPS 201 – SUMMER 2003

OUTLINE:

I. MATHEMATICAL PRELIMINARIES
- ASYMPTOTIC GROWTH RATES (CLRS 3)
- INDUCTION PROOFS (HANDOUT)
- RECURRENCE RELATIONS (CLRS 4)

II. DIVIDE & CONQUER ALGORITHMS
- SEARCHING
- SORTING (CLRS 7,8)
- SELECTION (CLRS 9)
- NON-COMPARISON SORTS (CLRS 8)

III. LOWER BOUNDS & COMPUTATIONAL COMPLEXITY
- DECISION TREES
- ADVERSARY ARGUMENTS

IV. DYNAMIC PROGRAMMING (CLRS 15)
- COIN CHANGING PROBLEM
- 0-1 KNAPSACK
- MATRIX CHAIN MULTIPLICATION, APSP

V. GREEDY ALGORITHMS (CLRS 16)
- CONTINUOUS KNAPSACK
- MINIMUM WEIGHT SPANNING TREES
- MATROIDS

VI. AMMORTIZED ANALYSIS (CLRS 17)
- FIBONACCI HEAPS (CLRS 20)
- DISJOINT SETS (CLRS 21)

VII. NP-COMPLETENESS (CLRS 34)

3.2 STANDARD NOTATION & FUNCTIONS.

FLOORS & CEILINGS

GIVEN $x \in \mathbb{R}$, $\lfloor x \rfloor$ AND $\lceil x \rceil$ ARE THE UNIQUE INTEGERS SATISFYING:

$$x-1 < \lfloor x \rfloor \leq x \leq \lceil x \rceil < x+1$$

EQUIVALENTLY:

(i) $N = \lfloor x \rfloor \iff N \leq x < N+1$

(ii) $N = \lceil x \rceil \iff N-1 < x \leq N$

THEOREM.
FOR ANY POSITIVE INTEGERS n, $a \neq 0$, $b \neq 0$.

(1) $\lfloor \lfloor n/a \rfloor / b \rfloor = \lfloor n/ab \rfloor$

(2) $\lceil \lceil n/a \rceil / b \rceil = \lceil n/ab \rceil$

WE PROVE (1). FIRST

LEMMA: GIVEN $x \geq 0$ REAL, $m > 0$ INTEGER

$$\lfloor \lfloor x \rfloor / m \rfloor = \lfloor x/m \rfloor .$$

((1) Follows by letting $x = \frac{n}{a}$, $m = b$.)

PROOF (OF LEMMA)

LET $N = \lfloor \frac{x}{m} \rfloor$. THEN $N \le \frac{x}{m} < N+1$, AND

$$N \cdot m \le x < (N+1)m$$

NOW $Nm \le x$ IMPLIES $Nm \le \lfloor x \rfloor$. ALSO $\lfloor x \rfloor \le x < (N+1)m$. THUS

$$Nm \le \lfloor x \rfloor < (N+1)m$$

$$\therefore \quad N \le \frac{\lfloor x \rfloor}{m} < N+1$$

$$\therefore \quad N = \left\lfloor \frac{\lfloor x \rfloor}{m} \right\rfloor . \qquad ///.$$

EXCERCISE: PROVE (2).

LOGARITHMS

$$x = a^{\log_a(x)} = \left(b^{\log_b(a)}\right)^{\log_a(x)} = b^{\log_b(a) \cdot \log_a(x)}$$

$$(*) \quad \therefore \quad \log_b(x) = \log_b(a) \cdot \log_a(x)$$

$$\therefore \quad \log_b(n) = \text{CONST} \cdot \log_a(n)$$

$\therefore$ $\boxed{\log_b(n) = \Theta(\log_a(n))}$

$$\therefore \quad \log_a(x) = \frac{\log_b(x)}{\log_b(a)}$$

$$\therefore \quad \lg(x) = \frac{\ln(x)}{\ln(2)}$$

EQUATION (*) IMPLIES

$$a^{\log_b(x)} = a^{\log_a(x)\cdot\log_b(a)} = \left(a^{\log_a(x)}\right)^{\log_b(a)} = x^{\log_b(a)}$$

$\therefore$ $\boxed{a^{\log_b(x)} = x^{\log_b(a)}}$

STIRLING'S FORMULA

$$n! = \sqrt{2\pi n}\cdot\left(\frac{n}{e}\right)^n\cdot\left(1+\Theta\left(\frac{1}{n}\right)\right)$$

COROLLARY

(i) $n! = o(n^n)$

(ii) $n! = \omega(2^n)$

(iii) $\lg n! = \Theta(n \lg n)$

PROOF OF (i)

$$\frac{n!}{n^n} = \frac{\sqrt{2\pi n}\cdot\left(\frac{n}{e}\right)^n\cdot\left(1+\Theta\left(\frac{1}{n}\right)\right)}{n^n} = \frac{\sqrt{2\pi n}\cdot\left(1+\Theta\left(\frac{1}{n}\right)\right)}{e^n} \to 0$$

AS $n \to \infty$.

$\therefore\ n! = o(n^n)$. ///

PROOF OF (iii)

$$\log n! = \log\sqrt{2\pi n} + \log\left(\frac{n}{e}\right)^n + \log\left(1+\Theta\left(\frac{1}{n}\right)\right)$$

$$= \frac{1}{2}\log 2\pi + \frac{1}{2}\log n + n\log n - n\log e + \log\left(1+\Theta\left(\frac{1}{n}\right)\right)$$

$\therefore$

$$\frac{\log n!}{n\log n} = 1 + (\text{STUFF WHICH} \to 0 \text{ AS } n\to\infty)$$

$\therefore\ \log n! = \Theta(n\log n)$. ///

EXCERCISE: PROVE (ii).

EXCERCISE: SHOW $\binom{2n}{n} = \Theta\left(\frac{4^n}{\sqrt{n}}\right)$.

RECALL $\binom{m}{k} = \frac{m!}{k!\,(m-k)!}$

- ASYMPTOTIC GROWTH RATES
- 3.2 : FLOOR & CEILING, logs, STIRLINGS FORMULA
- INDUCTION
  - o EXAMPLES 2, 3
  - o EXAMPLE $T(n) = \begin{cases} 0 & n = 1 \\ T(\lfloor n/2 \rfloor) + 1 & n \geq 2 \end{cases}$
  - o EXAMPLES A, B (INDUCTION TRAP)

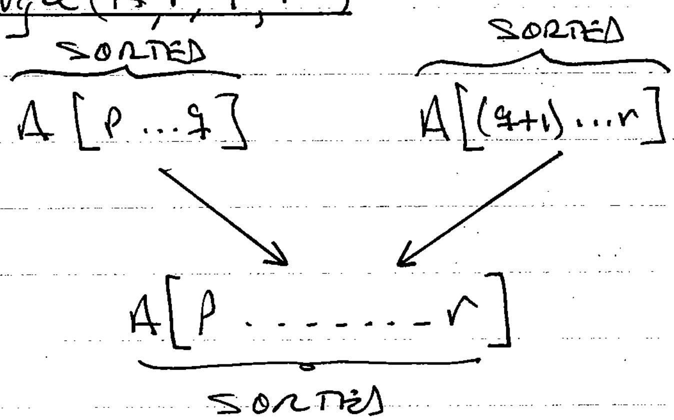
$\left\{ \begin{array}{l} \text{show } T(n) \leq \lg n \ \ \forall n \geq 1 \\ \therefore T(n) = O(\lg n). \end{array} \right.$

4 Recurrences

Consider the problem of sorting an array A. Let $A[p \ldots q]$ denote the subarray $(A_p, A_{p+1}, \ldots, A_q)$.

MergeSort (A, p, r)	Cost
1.) if $p < r$	$\Theta(1)$
2.) $\quad q \leftarrow \lfloor \frac{p+r}{2} \rfloor$	$\Theta(1)$
3.) $\quad$ MergeSort (A, p, q)	$T(\lceil \frac{n}{2} \rceil)$
4.) $\quad$ MergeSort $(A, q+1, r)$	$T(\lfloor \frac{n}{2} \rfloor)$
5.) $\quad$ Merge (A, p, q, r)	$\Theta(n)$

Merge (A, p, q, r)



Cost $= \Theta(n)$

$n = r - p + 1$

MergeSort divides the subarray $A[p \ldots r]$ (roughly) in half, recursively sorts the two halves, then merges the two sorted subarrays.