ALGORITHM DESIGN
DIVIDE AND CONQUER

by

David Toman

University of Waterloo

List of Slides

Divide and Conquer

IDEA: split an problem into smaller subproblems, (recursively) solve the subproblems, combine results

Examples:

- Quick-sort, Merge-Sort
- Multiplying large integers/polynomials
- Matrix Multiplication
- Searching

Integer Multiplication

Problem: multiply two *large integers*

$\Rightarrow$ naive algorithm: $O(n^2)$

Divide-and-Conquer approach:

Let $X = X_L \cdot 10^k + X_R$ and $Y = Y_L \cdot 10^k + Y_R$ and

1. $A = X_L Y_L$
2. $C = X_R Y_R$
3. $B = (X_L - Y_R)(Y_R - X_L) + A + C$

Then

$$X \cdot Y = 10^{2k} A + 10^k B + C$$

Complexity

The individual steps:

1. $T(N/2)$
2. $T(N/2)$
3. $2\Theta(N/2) + T(N/2) + 2O(N) = T(N/2) + \Theta(N)$

final sum: $O(N)$

$$T(N) = 3T(N/2) + \Theta(N)$$

and therefore $T(N) \in O(N^{\log_2 3}) = O(N^{1.59})$

Similarly:
Multiplying matrices $O(N^{\log_2 7})$ (Strassen's Algorithm)

Recurrences

Theorem: The solution to

$$T(N) = aT(N/b) + \Theta(N^k)$$

for $a \geq 1$ and $b > 1$ is

$$T(N) = \begin{cases} O(N^{\log_b a}) & for\ a > b^k \\ O(N^k \log N) & for\ a = b^k \\ O(N^k) & for\ a < b^k \end{cases}$$

Why?

- Guess-and-verify (hard)
- Constructive induction (telescoping sums)
- Characteristic polynomials

plus “smoothness rule”

⇒ cf. Graham, Knuth, Patashnik: Discrete Math.