SNS COLLEGE OF TECHNOLOGY

Coimbatore-35
An Autonomous Institution

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A+' Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

DEPARTMENT OF INFORMATION TECHNOLOGY

19ITB201 – DESIGN AND ANALYSIS OF ALGORITHMS

II YEAR IV SEM

UNIT-I-Introduction

TOPIC: Fundamentals of the Analysis of Algorithm Efficiency – Mathematical analysis for Recursive algorithms

Prepared by
C.PARKAVI,AP/AIML

MATHEMATICAL ANALYSIS FOR RECURSIVE ALGORITHM

Subject :Design and Analysis of Algorithm

Unit :I

- Analysis Framework
- Asymptotic Notations and its properties
- Mathematical analysis for Recursive algorithms.
- Mathematical analysis for Nonrecursive algorithms.

What is a recursive algorithm?

Example: Factorial

$n! = 1 \bullet 2 \bullet 3 \ldots n$ and $0! = 1$ (called initial case)

So the recursive defintiion $n! = n \bullet (n-1)!$

Algorithm *F*(*n*)

if n = 0 **then** return 1 // base case

else F(*n*-1)•*n* // recursive call

Basic operation? multiplication during the recursive call

Formula for multiplication

$M(n) = M(n\text{-}1) + 1$

is a recursive formula too. This is typical.

We need the initial case which corresponds to the base case

$M(0) = 0$

There are no multiplications

Solve by the method of *backward substitutions*

$M(n) = M(n\text{-}1) + 1$

$= [M(n\text{-}2) + 1] + 1 = M(n\text{-}2) + 2$ substituted $M(n\text{-}2)$ for $M(n\text{-}1)$

$= [M(n\text{-}3) + 1] + 2 = M(n\text{-}3) + 3$ substituted $M(n\text{-}3)$ for $M(n\text{-}2)$

... a pattern evolves

$= M(0) + n$

$= n$

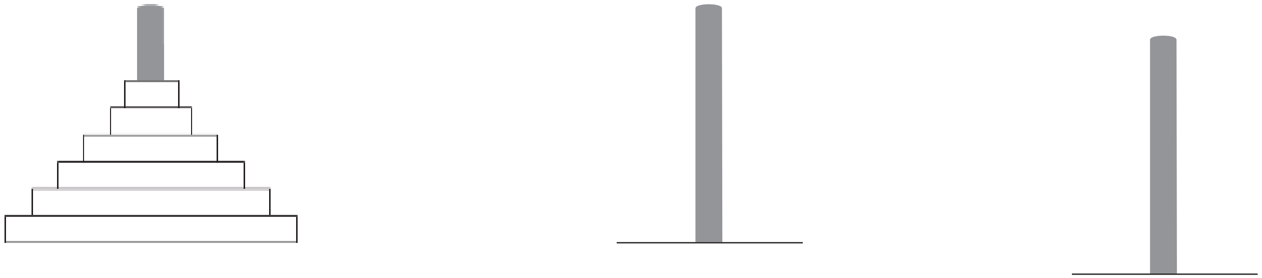
Not surprising!

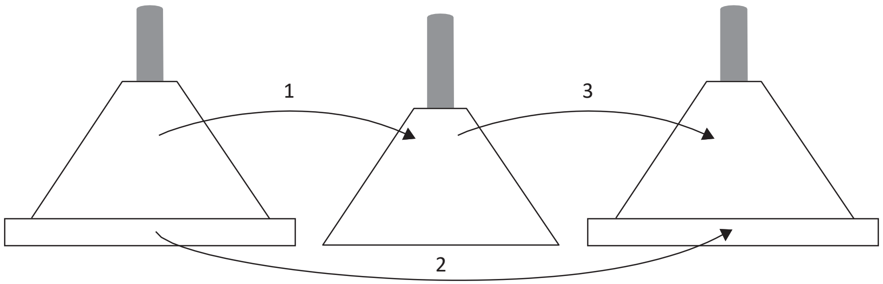
Therefore $M(n)\ \varepsilon\ \Theta(n)$

General Plan for Analyzing the Time Efficiency of Recursive Algorithms

- Decide on a parameter (or parameters) indicating an input’s size.
- Identify the algorithm’s basic operation.
- Check whether the number of times the basic operation is executed can vary on different inputs of the same size; if it can, the worst-case, average-case, and be st-case efficiencies must be investigated separately.
- Set up a recurrence relation, with an appropriate initial condition, for the numb er of times the basic operation is executed.
- Solve the recurrence or, at least, ascertain the order of growth of its solution.

Example: Tower Hanoi





Tower of Hanoi

1. Problem size is n, the number of discs

2. The basic operation is moving a disc from rod to anoth
er

3. There is no worst or best case

4. Recursive relation for moving n discs

$M(n) = M(n\text{-}1) + 1 + M(n\text{-}1) = 2M(n\text{-}1) + 1$

IC: $M(1) = 1$

5. Solve using backward substitution

$M(n) = 2M(n-1) + 1$

$= 2[2M(n-2) + 1] + 1 = 2^2M(n-2) + 2+1$

$=2^2[2M(n-3) +1] + 2+1 = 2^3M(n-3) + 2^2 + 2 + 1$

...

$M(n) = 2^iM(n-i) + \sum_{j=0}^{-i}2^j = 2^iM(n-i) + 2^i-1$

...

$M(n) = 2^{n-1}M(n-(n-1)) + 2^{n-1}-1 = 2^{n-1}M(1) + 2^{n-1}-1 = 2^{n-1} + 2^{n-1}-1 = 2^n-1$

$M(n) \varepsilon \Theta(2^n)$

Terrible. Can we do it better?

- Where did the exponential term come from? Because two recursive calls are made. Suppose three recursive calls are made, what is the order of growth.

- Lesson learned: Be careful of the recursive algorithm, they can grow exponential.

- Especial if the problem size is measured by the level of the recursive tree and the operation count is total number of nodes

Example: Binary Representation

Algorithm *BinRec*(*n*)

if *n* = 1 **then return** 1

else return *BinRec*(floor(*n*/2)) + 1

1. Problem size is n

2. Basic operation is the addition in the recursive call

3. There is no difference between worst and best case

4. Recursive relation including initial conditions

$A(n) = A(\text{floor}(n/2)) + 1$

IC $A(1) = 0$

5. Solve recursive relation

The division and floor function in the argument of the recursive call mak es the analysis difficult.

We could make the variable substitution, $n = 2^k$, could get rid of the defin ition,

but the substitution skips a lot of values for n.

The **smoothness rule** (see appendix B) says that is ok.

Smoothness rule

$T(n)$ eventually non-decreasing and $f(n)$ be smooth {eventually non-decreasing and $f(2n)$ ε $\Theta(f(n))$}
if $T(n)$ ε $\Theta(f(n))$ for n powers of b then $T(n)$ ε $\Theta(f(n))$ for all n.

Works for O and Ω.

substitute $n = 2^k$ (also $k = \lg(n)$)
$A(2^k) = A(2^{k-1}) + 1$ and IC $A(2^0) = 0$

$A(2^k) = [A(2^{k-2}) + 1] + 1 = A(2^{k-2}) + 2$
$= [A(2^{k-3}) + 1] + 2 = A(2^{k-3}) + 3$
...
$= A(2^{k-i}) + i$
...
$= A(2^{k-k}) + k$
$A(2^k) = k$

Substitute back $k = \lg(n)$
$A(n) = \lg(n)$ ε $\Theta(\lg n)$

Thank you!