4.Divide-and-Conquer

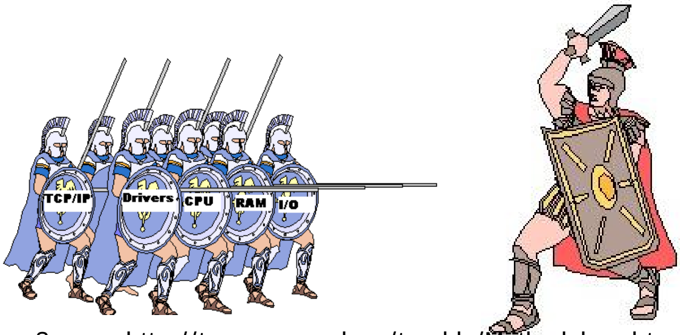
Chun-Chao Yeh

Acknowledgment
Prof. Lih-Hsing Hsu of NCTU slides for his Algorithm course.

Divide-and-Conquer

- The strategy
- Example
- How to analyze the complexity
  - Solving the recurrence function

Divide-and-Conquer



Source: http://teamapproach.ca/trouble/Methodology.htm

Source: http://freshread.files.wordpress.com/2009/07/ants-3.jpg

Examples:

- Merge-sort
- Find-max

$$T(n) = \begin{cases} \Theta(1), & \text{if } n = 1, \\ T(\lceil n/2 \rceil) + T(\lfloor n/2 \rfloor) + \Theta(n), & \text{if } n = 1 \end{cases}$$

$$T(n) = \begin{cases} \Theta(1), & \text{if } n = 1, \\ T(\lceil n/2 \rceil) + T(\lfloor n/2 \rfloor) + \Theta(1), & \text{if } n = 1 \end{cases}$$

$T(n) = T(\lceil n/2 \rceil) + T(\lfloor n/2 \rfloor) + \Theta(n)$

rewrite $T(n) = 2T(n/2) + \Theta(n)$ (for simplicity, igore n/2^ k could be an odd number)

4.1 the maximum-subarray problem

- Problem:
  - Given an array A[1:n], find a subarray A[i:j] such that summation of all elements in the subarray A[i:j]=A[i]+A[i+1]+…+A[j] is maximal, among all possible subarray.
- Example:
  - A=13,-3,-25,20,-3,-16,-23, 18,20,-7,12, -5,-22,15,-4,7
  - Max=18,20,-7,12

maximum-subarray:Algo(1)

- Algo:
  - Find-max-subarray(A,low,high)
    - Mid=(low+high)/2
    - Find-max-subArray(A,low,mid)
    - Find-max-subArray(A,mid+1,high)
    - Find-max-crossing-subarray(A,low,mid,high)

maximum-subarray:Algo(2)

- Find-max-crossing-subarray(A,low,mid,high)
  - lefSum=-inf; sum=0
  - For i=mid downto low
    - Sum+=A[i]
    - If (sum> leftSum) {leftSum=sum; maxLeft=i}
  - rightSum=-inf; sum=0
  - For j=mid+1 to hight
    - Sum+=A[j]
    - If (sum> rightSum) {rightSum; maxRight=j}
  - Return(maxLeft, maxRight, leftSum+rightSum)

Correctness and complexity

- Correctness: trivial
- Complexity
  - $T(n)=2T(n/2)+\theta(n)$
  - $\Rightarrow T(n)= \theta(n\lg n)$

4.2 Strassen's algorithm for matrix multiplication

$$A = (a_{ij})_{nxn}, B = (b_{ij})_{nxn}$$

$$C = A \cdot B = (c_{ij})_{nxn}, \quad c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj} \qquad complexity = \Theta(n^3)$$

A simple dvide-and conquer algorithm

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}, \ B = \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}, \ C = \begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix},$$

$$\begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \cdot \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}$$

$$C_{11} = A_{11} \cdot B_{11} + A_{12} \cdot B_{21}$$
$$C_{12} = A_{11} \cdot B_{12} + A_{12} \cdot B_{22}$$
$$C_{21} = A_{21} \cdot B_{11} + A_{22} \cdot B_{21}$$
$$C_{22} = A_{21} \cdot B_{12} + A_{22} \cdot B_{22}$$

$$T(n) = \begin{cases} \Theta(1), & \text{if } n = 1, \\ 8T(n/2) + \Theta(n^2), & \text{if } n > 1 \end{cases}$$

$$\Rightarrow T(n) = \Theta(n^3) \rightarrow \text{no faster!!}$$

However, with proper decomposition (and recombination)
we can show that 7 sub-block is enough (by Strassen's method)
and thus, $T(n) = 7T(n/2) + \Theta(n^2)$

$$\Rightarrow T(n) = \Theta(n^{\lg 7}) \rightarrow \text{faster!!}$$

Strassen's algorithm for matrix multiplication

- Details
  - Skips
  - Please refer to the context

4.3 The substitution method
: Mathematical induction

Recurrences -- $T(n) = aT(n/b) + f(n)$

- ***Substitution method***
- ***Recursion-tree method***
- ***Master method***

Technicalities

- We neglect certain technical details when we state and solve recurrences. A good example of a detail that is often glossed over is the assumption of integer arguments to functions. Boundary conditions is ignored. Omit floors, ceilings.

4.3 The substitution method : Mathematical induction

- The substitution method for solving recurrence entails two steps:

1. Guess the form of the solution.
2. Use mathematical induction to find the constants and show that the solution works.

Example

$$\begin{cases} T(n) = 2T(\lfloor n/2 \rfloor) + n \\ T(1) = 1 \end{cases}$$

(We may omit the initial condition later.)

Guess $T(n) = O(n \log n)$

Assume $T(\lfloor n/2 \rfloor) \le c\lfloor n/2 \rfloor \log\lfloor n/2 \rfloor$

$$T(n) \le 2(c\lfloor n/2 \rfloor \log\lfloor n/2 \rfloor) + n \le cn\log\frac{n}{2} + n$$
$$= cn\log n - cn\log 2 + n \le cn\log n \quad (\text{if } c \ge 1.)$$

Initial condition $1 = T(1) < cn\log 1 = 0(\rightarrow\leftarrow)$

However, $4 = T(2) < cn\log 2\ (\text{if } c \ge 4)$

- Making a good guess

$$T(n) = 2T(\lfloor n/2 \rfloor + 17) + n$$

We guess

$$T(n) = O(n\log n)$$

Making guess provides loose upper bound and lower bound. Then improve the gap.

Subtleties

$$T(n) = T(\lfloor n/2 \rfloor) + T(\lceil n/2 \rceil) + 1$$

Note:
1. Example: Find max./min.
2. show: ceil(n/2)+floor(n/2)=n
   proof: both cases are ok
   n=2k(even) or 2k+1 (odd)

- Guess $T(n) = O(n)$
- Assume $T(n) \le cn$

$$T(n) \le c\lfloor n/2 \rfloor + c\lceil n/2 \rceil + 1 \le cn + 1 \not\le cn$$

- However, assume $T(n) \le cn - b$

$$T(n) \le (c\lfloor n/2 \rfloor - b) + (c\lceil n/2 \rceil - b) + 1$$
$$\le cn - 2b + 1 \le cn - b \quad \text{(Choose } b \ge 1\text{)}$$

Show that the solution to $T(n) = 2T(\lfloor \frac{n}{2} \rfloor + 17) + n$ is $\boldsymbol{O}(n \lg n)$

Solution:

assume a > 0, b > 0, c > 0 and $T(n) \leqq an \lg n - b\lg n - c$

$$T(n) \leqq 2[(\tfrac{n}{2} + 17)\lg(\tfrac{n}{2} + 17) - b\lg(\tfrac{n}{2} + 17) - c] + n$$

$$\leqq (an + 34a)\lg(\tfrac{n}{2} + 17) - 2b\lg(\tfrac{n}{2} + 17) - 2c + n$$

$$\leqq an\lg(\tfrac{n}{2} + 17) + an\lg 2^{1/a} + (34a - 2b)\lg(\tfrac{n}{2} + 17) - 2c$$

$$\leqq an\lg(n)\, 2^{1/a} + (34a - 2b)\lg(n) - 2c$$

anlg(n) $2^{1/a}$+ (34a-2b)lg(n) - 2c

➔n≧ $\frac{n}{2}$ +17，n≧34

➔n≧ ($\frac{n}{2}$ +17) $2^{1/a}$，∵$2^{1/2}$≦1.5∴n ≧12

➔34a-2b≦ -b，b ≧34a

➔c > 0，-c > -2c

➔ T(n) ≦anlgn - blgn - c，T(n) ≦anlgn

➔ T(n) = O(nlgn)

Avoiding pitfalls

$$\begin{cases} T(n) = 2T(\lfloor n/2 \rfloor) + n \\ T(1) = 1 \end{cases}$$

- Assume $T(n) \le O(n)$
- Hence $T(n) \le cn$

  $T(n) \le 2(c\lfloor n/2 \rfloor) + n \le cn + n = O(n)$

  (Since c is a constant)

- *(WRONG!)* You cannot find such a *c*.

Changing variables

$$T(n) = 2T(\sqrt{n}) + \lg n$$

Let $m = \lg n$.

$$T(2^m) = 2T(2^{m/2}) + m$$

Then $S(m) = 2S(m/2) + m$.

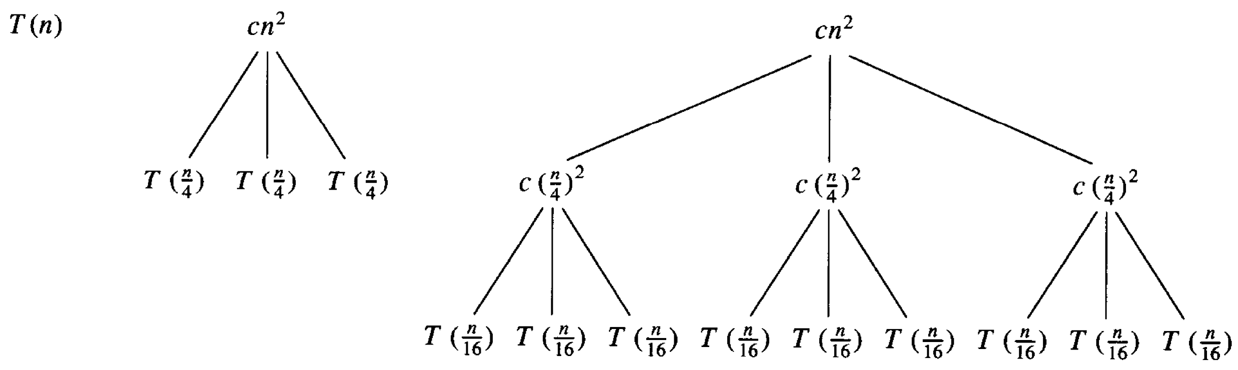
$$\Rightarrow S(m) = O(m \lg m)$$

$$\Rightarrow T(n) = T(2^m) = S(m) = O(m \lg m)$$

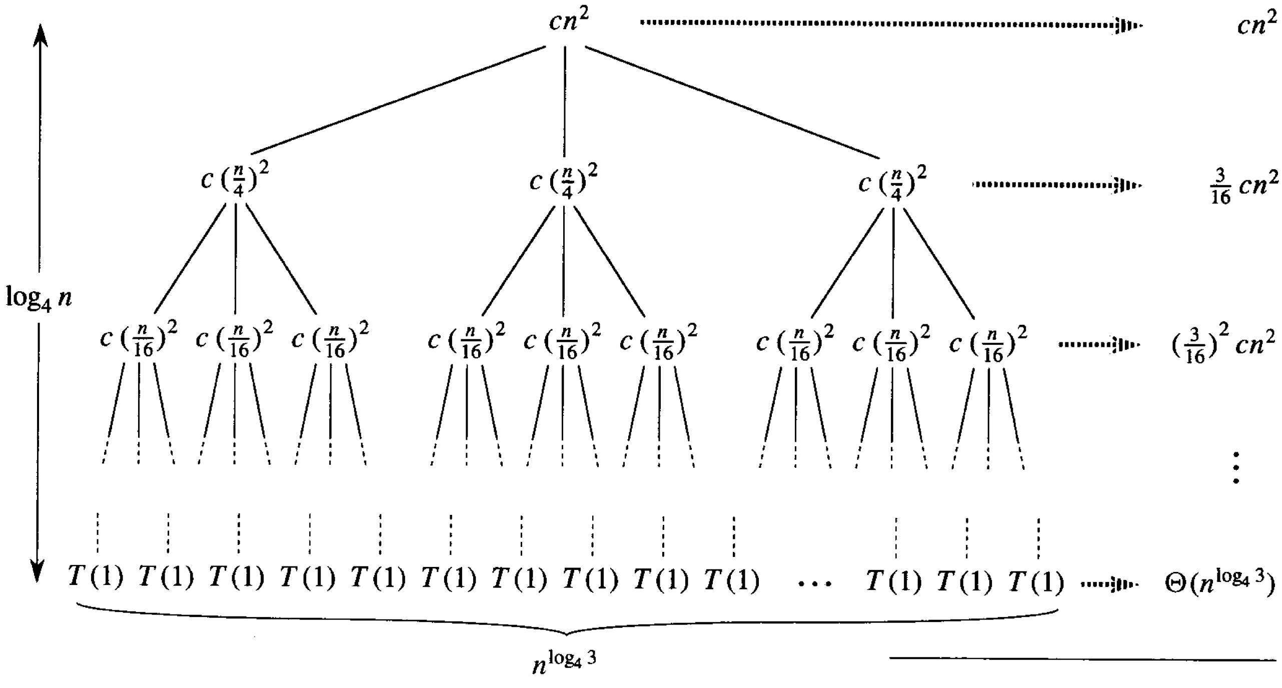
$$= O(\lg n \lg \lg n)$$

4.4 the Recursion-tree method

$$T(n) = 3T(\lfloor n/4 \rfloor) + \Theta(n^2)$$



(a) (b) (c)



(d)

Total: $O(n^2)$

The cost of the entire tree

$$T(n) = cn^2 + \frac{3}{16}cn^2 + \left(\frac{3}{16}\right)^2 cn^2 + \ldots + \left(\frac{3}{16}\right)^{\log_4 n - 1} cn^2 + \Theta(n^{\log_4 3})$$

$$= \sum_{i=0}^{\log_4 n - 1} \left(\frac{3}{16}\right)^i cn^2 + \Theta\left(n^{\log_4 3}\right)$$

$$= \frac{(3/16)^{\log_4 n} - 1}{(3/16) - 1} cn^2 + \Theta(n^{\log_4 3}).$$

$$T(n) = \sum_{i=0}^{\log_4 n-1} \left(\frac{3}{16}\right)^i cn^2 + \Theta\left(n^{\log_4 3}\right)$$

$$< \sum_{i=0}^{\infty} \left(\frac{3}{16}\right)^i cn^2 + \Theta\left(n^{\log_4 3}\right)$$

$$= \frac{1}{1-(3/16)} cn^2 + \Theta\left(n^{\log_4 3}\right)$$

$$= \frac{16}{13} cn^2 + \Theta(n^{\log_4 3})$$

$$= O(n^2)$$

substitution method

We want to Show that T(n) ≤ dn^2 for some constant d > 0. using the same constant c > 0 as before, we have

$$T(n) \le 3T(\lfloor n/4 \rfloor) + cn^2$$

$$\le 3d\lfloor n/4 \rfloor^2 + cn^2$$

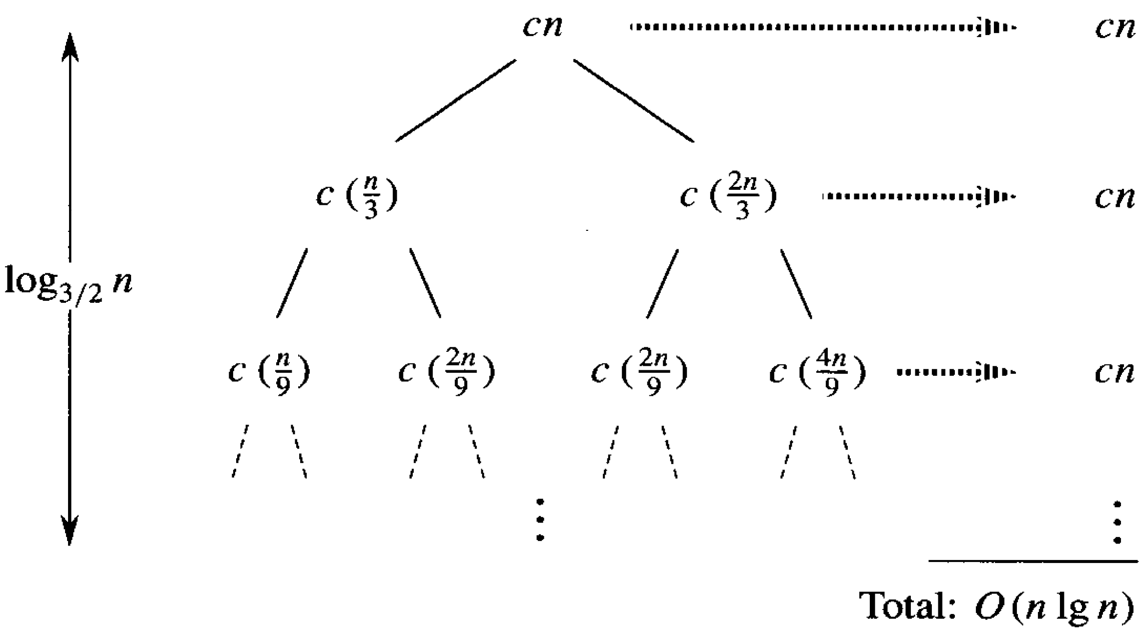
$$\le 3d(n/4)^2 + cn^2$$

$$= \frac{3}{16} dn^2 + cn^2$$

$$\le dn^2,$$

Where the last step holds as long as $d \geq (16/13)c$.

$$T(n) = T(n/3) + T(2n/3) + cn$$



substitution method

$$
\begin{aligned}
T(n) &\le T(n/3) + T(2n/3) + cn \\
&\le d(n/3)\lg(n/3) + d(2n/3)\lg(2n/3) + cn \\
&= (d(n/3)\lg n - d(n/3)\lg 3) + (d(2n/3)\lg n - d(2n/3)\lg(3/2)) + cn \\
&= dn\lg n - d((n/3)\lg 3 + (2n/3)\lg(3/2)) + cn \\
&= dn\lg n - d((n/3)\lg 3 + (2n/3)\lg 3 - (2n/3)\lg 2 + cn \\
&= dn\lg n - dn(\lg 3 - 2/3) + cn \\
&\le dn\lg n,
\end{aligned}
$$

As long as d $\geq$ c/lg3 – (2/3)).

4.5 The master method

Theorem 4.1 ***(Master theorem)***

Let $a \geq 1$ and $b > 1$ be constants, let $f(n)$ be a function, and $T(n)$ be defined on the nonnegative integers by the recurrence

$$T(n) = aT(n/b) + f(n)$$

where we interpret n/b mean either $\lfloor n/b \rfloor$ or $\lceil n/b \rceil$.

1. If $f(n) = O(n^{\log_b a - \varepsilon})$ for some constant $\varepsilon > 0$, then $T(n) = \Theta(n^{\log_b a})$.
2. If $f(n) = \Theta(n^{\log_b a})$ then $T(n) = \Theta(n^{\log_b a} \log n)$.
3. If $f(n) = \Omega(n^{\log_b a + \varepsilon})$ for some constant $\varepsilon > 0$ and if $af(n/b) = cf(n)$ for some constant $c < 1$ and all sufficiently large n, then $T(n) = \Theta(f(n))$.

Proof. (In section 4.4 by recursive tree)

- $T(n) = 9T(n/3) + n$

$a = 9, b = 3, f(n) = n$

$n^{\log_3 9} = n^2, \quad f(n) = O(n^{\log_3 9 - 1})$

$Case \quad 1 \Rightarrow T(n) = \Theta(n^2)$

- $T(n) = T(2n/3) + 1$

$a = 1, b = 3/2, f(n) = 1$

$n^{\log_{3/2} 1} = n^0 = 1 = f(n),$

$Case \quad 2 \Rightarrow T(n) = \Theta(\log n)$

- $T(n) = 3T(n/4) + n\log n$

$a = 3, b = 4, f(n) = n\log n$

$n^{\log_4 3} = n^{0.793}, \quad f(n) = O(n^{\log_4 3+\varepsilon})$

Case 3

Check

$af(n/b) = 3(\frac{n}{4})\log(\frac{n}{4}) \le \frac{3n}{4}\log n = cf(n)$

for $c = \frac{3}{4}$, and sufficiently large n

$\Rightarrow T(n) = \Theta(n\log n)$

User of master theorem--NOTE!!!! (1)

1.if $(n) = O(n^{\log_b a-\varepsilon})$ for some const. $\varepsilon > 0$, then $T(n) = \Theta(n^{\log_b a})$,

In the first case,
not only must f(n) be smaller than $n^{\log_b a}$, it must be *polynomially* smaller.
That is, f(n) must be asymptotically smaller than $n^{\log_b a}$ by a factor of n^{ε} .

3.if $(n) = \Omega(n^{\log_b a+\varepsilon})$ for some const. $\varepsilon > 0$, and if $af(n/b) \le cf(n)$
for some const. $c > 1$ and sufficiently large n, then $T(n) = \Theta(f(n))$.

In the third case,
Not only must f(n) be larger than $n^{\log_b a}$, it must be *polynomially* larger.
And satisfy the "regularity" condition that af(n/b) <= cf(n).

User of master theorem--NOTE!!!! (2)

Note that the three cases do not cover all the possiblities for f(n).
There is a gap between cases 1 and 2 when f(n) is smaller than $n^{\log_b a}$
But not polynomially smaller. Similarly,

there is a gap between cases 2 and 3 when f(n) is larger than $n^{\log_b a}$
But not polynomially larger.

If the function f(n) falls into one of these gaps, or if the regularity Condition in case 3 fails to hold, you cannot use the master method.

User of master theorem--NOTE!!!! (3)

Example:

The master method does not apply to the recurrence

$T(n) = 2T(n/2) + n\lg n,$

even though it has the proper form:

$a = 2, b = 2, f(n) = n\lg n, n^{\log_b a} = n.$

It might seem that case 3 should apply, since

$f(n) = n\lg n$ is asymptotically larger than $n^{\log_b a} = n$

The problem is that it is not polynomially larger

(by a factor of n^{ε}).

The ratio $f(n)/n^{\log_b a} = n\lg n / n = \lg n$ is asymptotically less than n^{ε} for any positive constant ε.
Consequently, the recurrence falls into the gap between case 2 and case 3.
(solution. see exercise 4.6-2)

Exercises: 4.5-5

Exercises : 4.5 - 5

consider the regularity condition $af(n/b) \le cf(n)$ for some constant c > 1, which is part of case 3 of the master theorem. Give an example of constant $a \ge 1$ anb $b > 1$ and a function $f(n)$ that satisfies all the conditions in case 3 of the master theorem except the gegularity condition.

$af(n/b) \le cf(n) \wedge f(n) = \Omega(n^{\log_b a+\varepsilon})$

let $f(n) = n^{\log_b a+\varepsilon}$, we want

no c < 1 satisfies $af(n/b) \le cf(n)$

$af(n/b) = a(n/b)^{\log_b a+\varepsilon} = an^{\log_b a+\varepsilon}/b^{\log_b a+\varepsilon} = an^{\log_b a+\varepsilon}/ab^{\varepsilon} = n^{\log_b a+\varepsilon}/b^{\varepsilon}$

$= f(n)/b^{\varepsilon} = c' f(n)$ Wrong, no such b

so, if $c' = 1/b^{\varepsilon} \ge 1$, then we cannot find c < 1, $af(n/b) \le cf(n)$

Exercises: 4.6-2

Exercises : 4.6 - 2

show that if $f(n) = \Theta(n^{\log_b a} \lg^k n)$, where $k \ge 0$, then the master recurrence has solution $T(n) = \Theta(n^{\log_b a} \lg^{k+1} n)$. (for simplicity, confine your analysis to exact powers of b.)

$$T(n) = \Theta(n^{\log_b a}) + \sum_{j=0}^{\log_b n - 1} a^j f(n/b^j) \text{(from Lemma 4.2)}$$

$$\text{let } g(n) = \sum_{j=0}^{\log_b n - 1} a^j f(n/b^j) = \Theta\left(\sum_{j=0}^{\log_b n - 1} a^j (n/b^j)^{\log_b a} \lg^k (n/b^j)\right)$$

$$= \Theta\left(\sum_{j=0}^{\log_b n - 1} a^j (n^{\log_b a} / (b^j)^{\log_b a}) \lg^k (n/b^j)\right)$$

$$= \Theta\left(n^{\log_b a} \sum_{j=0}^{\log_b n - 1} a^j (1/a^j) \lg^k (n/b^j)\right) = \Theta\left(n^{\log_b a} \sum_{j=0}^{\log_b n - 1} \lg^k (n/b^j)\right)$$

assume $\mathrm{n} = \mathrm{b}^{\mathrm{i}}$

$$\Rightarrow g(n) = \Theta\left(n^{\log_b a} \sum_{j=0}^{\log_b n - 1} \lg^k (b^i / b^j)\right) = \Theta\left(n^{\log_b a} \sum_{j=0}^{i-1} \lg^k b^{i-j}\right)$$

$$= \Theta\left(n^{\log_b a} \sum_{j=1}^{i} \lg^k b^j\right) = \Theta\left(n^{\log_b a} \sum_{j=1}^{i} (j \lg b)^k\right)$$

$$= \Theta\left(n^{\log_b a} \lg^k b \sum_{j=1}^{i} j^k\right)$$

$$\text{let } \mathrm{h(i,k)} = \sum_{j=1}^{i} j^k = 1^k + 2^k + \ldots + i^k < i^k + i^k + \ldots + i^k = i i^k = i^{k+1} = (\log_b n)^{k+1}$$

$$\text{show that } \mathrm{h(i,k)} > (\mathrm{i}/2)^{k+1} \left(\because (x^k + y^k) \geq \left(\frac{x+y}{2}\right)^k\right)$$

Exercises: 4.6-3

Exercises : 4.6 - 3

show that case 3 of the master theorem is overstated., in the sense that the regularity condition $af(n/b) \leq cf(n)$ for some constant $c < 1$ implies that there exists a constant $\varepsilon > 0$ such that $f(n) = \Omega(n^{\log_b a + \varepsilon})$.

$idea:$

$let\ n = \Theta(b^k), k = \Theta(\log_b n)$

$af(n/b) \leq cf(n) \Rightarrow f(n) \geq (a/c) f(n/b) \geq (a/c)^2 f(n/b^2)$

$\ldots. \geq (a/c)^k f(n/b^k) = (a/c)^{\Theta(\log_b n)} f(\Theta(1))$

$= \Theta(n^{\log_b a} . n^{-\log_b c}) f(\Theta(1)) = \Theta(n^{\log_b a} . n^{-\log_b c})$ (since $f(\Theta(1)) = \Theta(1)$)

$\because c$ is some constant < 1, $1/\mathrm{c} > 1$, $\log_\mathrm{b}(1/c) > 0$,

$\exists$ a constant $\varepsilon > 0$ such that $\varepsilon > \log_\mathrm{b}(1/c) > 0$

$\Rightarrow f(n) \geq \Theta(n^{\log_b a} . n^{-\log_b c}) \geq \Theta(n^{\log_b a} . n^{\varepsilon}) = \Theta(n^{\log_b a + \varepsilon})$

$\Rightarrow f(n) = \Omega(n^{\log_b a + \varepsilon})$

4.6 proof of the master theorem

- Lemma 4.2:
  - Let a ≧ 1 and b > 1 be constants, and let f(n) be a nonnegative function defined on exact powers of b. Define T(n) on exact powers of b by the recurrence

$$T(n)=\begin{cases}\Theta(1), & n=1\\ aT(n/b)+f(n), & n=b^i\end{cases}$$

where i is a positive integer. Then

$$T(n)=\Theta(n^{\log_b^a})+\sum_{j=0}^{\log_b^n-1}a^j f(n/b^j).$$

Proof: Lemma 4.2

$$\mathrm{T}(n)=aT(n/b)+f(n),\quad n=b^i$$

$$i=\log_b^n$$
$$a^{\log_b^n}=n^{\log_b^a},$$
$$S(0)=\mathrm{T}(b^0)=\mathrm{T}(1)=C\ (\text{constant})$$

By definition

$$\mathrm{T}(b^i)=aT(b^i/b)+f(b^i)=aT(b^{i-1})+f(b^i)$$

Changing variables

$$let \quad S(i)=T(b^i)$$

$$\begin{aligned}&\Rightarrow S(i)=aS(i-1)+f(b^i)=a(aS(i-2)+f(b^{i-1}))+f(b^i)\\&=a^2S(i-2)+af(b^{i-1})+f(b^i)\\&=a^2(aS(i-3)+f(b^{i-2}))+af(b^{i-1})+f(b^i)\\&=a^3S(i-3)+a^2f(b^{i-2}))+af(b^{i-1})+f(b^i)\\&=\ldots\\&=a^iS(0)+(a^{i-1}f(b^1)+\ldots+af(b^{i-1})+f(b^i))\end{aligned}$$

$$\Theta(n^{\log_b^a})$$

$$\Rightarrow \mathrm{T}(n)=T(1)(n^{\log_b^a})+\sum_{j=0}^{\log_b^n-1}a^j f(n/b^j).$$

4.6 proof of the master theorem (2)

- Lemma 4.3:
  - Let a ≧ 1 and b > 1 be constants, and let f(n) be a nonnegative function defined on exact powers of b. A function g(n) defined over exact powers of by

$$g(n) = \sum_{j=0}^{\log_b^n - 1} a^j f(n/b^j)$$

has the following asymptotic bounds for exact powers of b:

$$g(n) = \begin{cases} O(n^{\log_b^a}), & \text{if } f(n) = O(n^{\log_b^a - \varepsilon}) \text{ for some constant } \varepsilon > 0 \\ \Theta(n^{\log_b^a} \lg n), & \text{if } f(n) = \Theta(n^{\log_b^a}) \\ \Theta(f(n)), & \text{if } af(n/b) \le cf(n) \text{ for some constant } c < 1 \\ & \text{and for all sufficiently large n} \end{cases}$$

Proof: Lemma 4.3

given $n = b^i$

$$g(n) = \sum_{j=0}^{\log_b^n - 1} a^j f(n/b^j)$$

$$g(n) = a^{k-1} f(b^1) + \ldots + af(b^{k-1}) + f(b^k)$$

1. $g(n) = O(n^{\log_b^a})$, if $f(n) = O(n^{\log_b^a - \varepsilon})$ for some constant $\varepsilon > 0$

2. $g(n) = \Theta(n^{\log_b^a} \lg n)$, if $f(n) = \Theta(n^{\log_b^a})$

3. $g(n) = \Theta(f(n))$, if $af(n/b) \le cf(n)$ for some constant $c < 1$ and for all sufficiently large n

Proof: Lemma 4.3(2)

$$1.\ g(n) = O(n^{\log_b a}), \text{if } f(n) = O(n^{\log_b a - \varepsilon}) \text{ for some constant } \varepsilon > 0$$

given $n = b^k$

$$g(n) = \sum_{j=0}^{\log_b n - 1} a^j f(n/b^j)$$

$$g(n) = a^{k-1} f(b^1) + \ldots + af(b^{k-1}) + f(b^k)$$

if $f(n) = O(n^{\log_b a - \varepsilon})$

$$\Rightarrow g(n) = \sum_{j=0}^{\log_b n - 1} a^j f(n/b^j) = O\left(\sum_{j=0}^{\log_b n - 1} a^j (n/b^j)^{\log_b a - \varepsilon}\right)$$

$$\sum_{j=0}^{\log_b n - 1} a^j (n/b^j)^{\log_b a - \varepsilon} = n^{\log_b a - \varepsilon}\left(\sum_{j=0}^{\log_b n - 1} a^j (b^{-j})^{\log_b a - \varepsilon}\right)$$

$$= n^{\log_b a - \varepsilon}\left(\sum_{j=0}^{\log_b n - 1} a^j (b^{-j\log_b a})(b^{j\varepsilon})\right) = n^{\log_b a - \varepsilon}\left(\sum_{j=0}^{\log_b n - 1} a^j (a^{-j})(b^{j\varepsilon})\right)$$

$$= n^{\log_b a - \varepsilon}\left(\sum_{j=0}^{\log_b n - 1} b^{j\varepsilon}\right) = n^{\log_b a - \varepsilon}\left(\frac{(b^\varepsilon)^{\log_b n} - 1}{b^\varepsilon - 1}\right) = n^{\log_b a - \varepsilon}\left(\frac{n^\varepsilon - 1}{b^\varepsilon - 1}\right) < n^{\log_b a}$$

Proof: Lemma 4.3(3)

given $n = b^k$

$$g(n) = \sum_{j=0}^{\log_b n - 1} a^j f(n/b^j)$$

$$g(n) = a^{k-1} f(b^1) + \ldots + af(b^{k-1}) + f(b^k)$$

$$2.\ g(n) = \Theta(n^{\log_b a} \lg n), \text{if } f(n) = \Theta(n^{\log_b a})$$

if $f(n) = \Theta(n^{\log_b a})$

$$\Rightarrow g(n) = \sum_{j=0}^{\log_b n - 1} a^j f(n/b^j) = \Theta\left(\sum_{j=0}^{\log_b n - 1} a^j (n/b^j)^{\log_b a}\right)$$

$$\sum_{j=0}^{\log_b n - 1} a^j (n/b^j)^{\log_b a} = n^{\log_b a}\left(\sum_{j=0}^{\log_b n - 1} a^j (b^{-j})^{\log_b a}\right)$$

$$= n^{\log_b a}\left(\sum_{j=0}^{\log_b n - 1} a^j (b^{-j\log_b a})\right) = n^{\log_b a - \varepsilon}\left(\sum_{j=0}^{\log_b n - 1} a^j (a^{-j})\right)$$

$$= n^{\log_b a}\left(\sum_{j=0}^{\log_b n - 1} 1\right) = (n^{\log_b a})(\log_b n)$$

Proof: Lemma 4.3(4)

3. $g(n) = \Theta(f(\mathrm{n}))$, if $af(n/b) \le cf(n)$ for some constant $c < 1$
and for all sufficiently large n

if $af(n/b) \le cf(n)$

$$\Rightarrow g(n) = \sum_{j=0}^{\log_b^n - 1} a^j f(n/b^j) \le \sum_{j=0}^{\log_b^n - 1} a^{j-1}(cf(n/b^{j-1}))$$

$$\le \sum_{j=0}^{\log_b^n - 1} a^{j-2}(c^2 f(n/b^{j-2}))$$

....

$$\le \sum_{j=0}^{\log_b^n - 1} a^{j-j}(c^j f(n/b^{j-j})) \le \sum_{j=0}^{\log_b^n - 1} c^j f(n) = f(n) \sum_{j=0}^{\log_b^n - 1} c^j \le f(n)\sum_{j=0}^{\infty} c^j$$

$$\le f(n)(\frac{1}{1-c}) \text{ (since } c < 1)$$

$$= O(f(n))$$

given $n = b^k$

$$g(n) = \sum_{j=0}^{\log_b^n - 1} a^j f(n/b^j)$$

$$g(n) = a^{k-1} f(b^1) + \ldots + af(b^{k-1}) + f(b^k)$$

given $n = b^k$

$$g(n) = \sum_{j=0}^{\log_b^n - 1} a^j f(n/b^j)$$

$$= f(n) + \sum_{j=1}^{\log_b^n - 1} a^j f(n/b^j) \ge f(n)$$

$$\Rightarrow g(n) = \Omega(f(n))$$

$$g(n) = O(f(n)) \wedge \Omega(f(n))$$
$$\Rightarrow g(n) = \Theta(f(n))$$

4.6 proof of the master theorem

- Lemma 4.4:
  - Let a ≧ 1 and b > 1 be constants, and let f(n) be a nonnegative function defined on exact powers of b. Define T(b) on exact powers of b by the recurrence

$$T(n) = \begin{cases} \Theta(1), & n = 1 \\ aT(n/b) + f(n), & n = b^i \end{cases}$$

where i is a positive integer. Then T(n) has the following asymptotic bounds for exact powers of b:

$$\mathrm{T}(n) = \begin{cases} \Theta(\mathrm{n}^{\log_b^a}), & \text{if } \mathrm{f(n)} = \mathrm{O}(\mathrm{n}^{\log_b^a - \varepsilon}) \text{ for some constant } \varepsilon > 0 \\ \Theta(\mathrm{n}^{\log_b^a} \lg \mathrm{n}), & \text{if } \mathrm{f(n)} = \Theta(\mathrm{n}^{\log_b^a}) \\ \Theta(f(\mathrm{n})), & \text{if } \mathrm{f(n)} = \Omega(\mathrm{n}^{\log_b^a + \varepsilon}) \text{ for some constant } \varepsilon > 0, \\ & \text{and if } af(n/b) \le cf(n) \text{ for some constant c} < 1 \\ & \text{and for all sufficiently large n} \end{cases}$$

Proof: Lemma 4.4

- From lemma 4.2

$$T(n) = aT(n/b) + f(n), \quad n = b^i$$

$$T(n) = C(n^{\log_b a}) + \sum_{j=0}^{\log_b a - 1} a^j f(n/b^j).$$

$$= \Theta(n^{\log_b a}) + g(n)$$

- From lemma 4.3

$$g(n) = \begin{cases} O(n^{\log_b a}), & \text{if } f(n) = O(n^{\log_b a - \varepsilon}) \text{ for some constant } \varepsilon > 0 \\ \Theta(n^{\log_b a} \lg n), & \text{if } f(n) = \Theta(n^{\log_b a}) \\ \Theta(f(n)), & \text{if } af(n/b) \le cf(n) \text{ for some constant } c < 1 \\ & \text{and for all sufficiently large n} \end{cases}$$

We have

$$T(n) = \begin{cases} \Theta(n^{\log_b a}) + O(n^{\log_b a}) = \Theta(n^{\log_b a}), & \text{if } f(n) = O(n^{\log_b a - \varepsilon}) \text{ for some constant } \varepsilon > 0 \\ \Theta(n^{\log_b a}) + \Theta(n^{\log_b a} \lg n) = \Theta(n^{\log_b a} \lg n), & \text{if } f(n) = \Theta(n^{\log_b a}) \\ \Theta(n^{\log_b a}) + \Theta(f(n)) = \Theta(f(n)), & \text{if } f(n) = \Omega(n^{\log_b a + \varepsilon}) \text{ for some constant } \varepsilon > 0, \\ & \text{and if } af(n/b) \le cf(n) \text{ for some constant } c < 1 \\ & \text{and for all sufficiently large n} \end{cases}$$

Are we done?

- What if n is not exact powers of b?
- What if we have ceilings/floors?

What if n is not exact powers of b?

$$T(n) = \begin{cases} \Theta(1), & n = 1 \\ aT(n/b) + f(n), & n > 1, a \ge 1, b > 1 \end{cases}$$

$$T(n) = \begin{cases} \Theta(n^{\log_b^a}), & \text{if } f(n) = O(n^{\log_b^a - \varepsilon}) \text{ for some constant } \varepsilon > 0 \\ \Theta(n^{\log_b^a} \lg n), & \text{if } f(n) = \Theta(n^{\log_b^a}) \\ \Theta(f(n)), & \text{if } f(n) = \Omega(n^{\log_b^a + \varepsilon}) \text{ for some constant } \varepsilon > 0, \end{cases}$$

and if $af(n/b) \le cf(n)$ for some constant $c < 1$

and for all sufficiently large n

Note that:

$$T(n) = aT(n/b) + f(n), \quad n = b^i$$

$$T(n) = C(n^{\log_b^a}) + \sum_{j=0}^{\log_b^a - 1} a^i f(n/b^j).$$

let $b^k \le n < b^{k+1}$, for some $k \ge 1$

$$T(b^k) \le T(n) \le T(b^{k+1})$$

(assume T(n) is non - decreasing func.)

What if n is not exact powers of b?(2)

let $b^k \le n < b^{k+1}$, for some $k \ge 1$

$$T(b^k) \le T(n) \le T(b^{k+1})$$

$T(n) = \Theta(n^{\log_b^a})$, if $f(n) = O(n^{\log_b^a - \varepsilon})$ for some constant $\varepsilon > 0$

$T(n) = \Theta(n^{\log_b^a} \lg n)$, if $f(n) = \Theta(n^{\log_b^a})$

$T(n) = \Theta(f(n))$, if $f(n) = \Omega(n^{\log_b^a + \varepsilon})$ for some constant $\varepsilon > 0$,

and if $af(n/b) \le cf(n)$ for some constant $c < 1$

and for all sufficiently large n

What if n is not exact powers of b?(3)

- Proof: let $b^k \le n < b^{k+1}$, for some $k \ge 1$

  $T(b^k) \le T(n) \le T(b^{k+1})$

$$1. T(n) = \Theta(n^{\log_b^a}), \text{ if } f(n) = O(n^{\log_b^a - \varepsilon}) \text{ for some constant } \varepsilon > 0$$

$$T(n) \le T(b^{k+1}) = \Theta((b^{(k+1)\log_b^a})) = \Theta(a^{k+1}) \le c_1 a^{k+1} = c_1 a a^k \le c_1 a a^{\log_b^n} = c_1 a n^{\log_b^a} \ (\because b^k \le n)$$

$$\Rightarrow T(n) = O(n^{\log_b^a}) \ldots (1)$$

$$T(n) \ge T(b^k) = \Theta((b^{(k)\log_b^a})) = \Theta(a^k) \ge c_2 a^k = c_2 a^{-1} a^{k+1} \ge c_2 a^{-1} a^{\log_b^n} = c_2 a^{-1} n^{\log_b^a} \ (\because b^{k+1} > n)$$

$$\Rightarrow T(n) = \Omega(n^{\log_b^a}) \ldots (2)$$

$$(1),(2) \Rightarrow T(n) = \Theta(n^{\log_b^a})$$

What if n is not exact powers of b?(4)

- Proof: let $b^k \le n < b^{k+1}$, for some $k \ge 1$

  $T(b^k) \le T(n) \le T(b^{k+1})$

$$2. T(n) = \Theta(n^{\log_b^a} \lg n), \text{ if } f(n) = \Theta(n^{\log_b^a})$$

$$T(n) \le T(b^{k+1}) = \Theta((b^{(k+1)\log_b^a})) = \Theta(a^{k+1}) \le c_1 a^{k+1} = c_1 a a^k \le c_1 a a^{\log_b^n} = c_1 a n^{\log_b^a} \ (\because b^k \le n)$$

$$\Rightarrow T(n) = O(n^{\log_b^a}) \ldots (1)$$

$$T(n) \ge T(b^k) = \Theta((b^{(k)\log_b^a})) = \Theta(a^k) \ge c_2 a^k = c_2 a^{-1} a^{k+1} \ge c_2 a^{-1} a^{\log_b^n} = c_2 a^{-1} n^{\log_b^a} \ (\because b^{k+1} > n)$$

$$\Rightarrow T(n) = \Omega(n^{\log_b^a}) \ldots (2)$$

$$(1),(2) \Rightarrow T(n) = \Theta(n^{\log_b^a})$$

What if n is not exact powers of b?(5)

- Proof: let $b^k \le n < b^{k+1}$, for some $k \ge 1$

$$T(b^k) \le T(n) \le T(b^{k+1})$$

3. $T(n) = \Theta(f(n))$, if $f(n) = \Omega(n^{\log_b a + \varepsilon})$ for some constant $\varepsilon > 0$,
and if $af(n/b) \le cf(n)$ for some constant $c < 1$
and for all sufficiently large n

1. $T(n) = O(f(n))$

$T(n) = aT(n/b) + f(n) \le adf(n/b) + f(n)$

$\le dcf(n) + f(n) (\because af(n/b) \le cf(n))$

$= df(n) - (d(1-c)f(n) - f(n))$

$\le df(n)$ (if $d(1-c)f(n) \ge f(n)$

$=> d(1-c) \ge 1 => d \ge \dfrac{1}{1-c}$)

2. $T(n) = \Omega(f(n))$

trivial : by definition ,

$\Omega(g(n)) = \{f(n) \mid \exists c, n_0 \text{ s.t. } 0 \le cg(n) \le f(n)\ \forall n \ge n_0\} \Rightarrow f(n) = \Omega(g(n))$

so, we can easily find

$c = 1, n_0 = b,\ 0 \le cg(n) = f(n) \le aT(n/b) + f(n), \forall n \ge n_0$

4.6.2 floors and ceilings

- Some what complicate, refer to the textbook

Lower bound on

$$T(n) = aT(\lceil n/b \rceil) + f(n),$$

Upper bound on

$$T(n) = aT(\lfloor n/b \rfloor) + f(n),$$

$n/b \le \lceil n/b \rceil < n/b + 1,$

$\Rightarrow T(n/b) \le T(\lceil n/b \rceil) \le T(n/b+1)$

$n/b - 1 < \lfloor n/b \rfloor \le n/b,$

$\Rightarrow T(n/b-1) \le T(\lfloor n/b \rfloor) \le T(n/b)$

idea :

$\exists k, m/b = b^{k-1} \le \lceil n/b \rceil < b^k, (b > 1, k > 1)$

$\Rightarrow m/b = b^{k-1} \le n/b \le \lceil n/b \rceil < n/b + 1 < b^k, (b > 1, k > 1)$

$\Rightarrow m = b^k \le n < b^{k+1} = bm$

$T(b^{k-1}) \le T(\lceil n/b \rceil) \le T(b^k),$

we want : $\Omega(g(m)) \le T(\lceil n/b \rceil) \le O(g(m))$

$\Rightarrow T(\lceil n/b \rceil) = \Theta(g(m))$

$\because \Theta(g(m)) = \Theta(g(bm)) \wedge m \le n < bm$

$\Rightarrow \Theta(g(m)) = \Theta(g(bm)) = \Theta(g(n))$

floors and ceilings (2)

$$T(n) = \begin{cases} \Theta(1), & n = 1 \\ aT(n/b) + f(n), & n > 1, a \geq 1, b > 1 \end{cases}$$

$$T(n) = \begin{cases} \Theta(\mathrm{n}^{\log_b{}^a}), & \text{if } f(n) = O(\mathrm{n}^{\log_b{}^a - \varepsilon}) \text{ for some constant } \varepsilon > 0 \\ \Theta(\mathrm{n}^{\log_b{}^a} \lg \mathrm{n}), & \text{if } f(n) = \Theta(\mathrm{n}^{\log_b{}^a}) \\ \Theta(f(\mathrm{n})), & \text{if } f(n) = \Omega(\mathrm{n}^{\log_b{}^a + \varepsilon}) \text{ for some constant } \varepsilon > 0, \\ & \text{and if } af(n/b) \leq cf(n) \text{ for some constant } c < 1 \\ & \text{and for all sufficiently large n} \end{cases}$$

Note that:

$$n/b \leq \lceil n/b \rceil < n/b + 1,$$
$$\Rightarrow \mathrm{T}(n/b) \leq T(\lceil n/b \rceil) \leq T(n/b+1)$$
$$n/b - 1 < \lfloor n/b \rfloor \leq n/b,$$
$$\Rightarrow \mathrm{T}(n/b-1) \leq T(\lfloor n/b \rfloor) \leq T(n/b)$$

T(n)=aT(n/b+c)+f(n)?

- Recall Lemma 4.2!!
  - Lemma 4.2:
    - Let a ≧ 1 and b > 1 be constants, and let f(n) be a nonnegative function defined on exact powers of b. Define T(n) on exact powers of b by the recurrence

$$T(n) = \begin{cases} \Theta(1), & n = 1 \\ aT(n/b) + f(n), & n = b^i \end{cases}$$

where i is a positive integer. Then

$$T(n) = \Theta(\mathrm{n}^{\log_b{}^a}) + \sum_{\mathrm{j}=0}^{\log_b{}^n - 1} a^j f(n/b^j).$$

T(n)=aT(n/b+c)+f(n)? (2)

$$T(n)=\begin{cases}\Theta(1), & n=1\\ aT(n/b+c)+f(n), & n=b^i\end{cases}$$

where i is a positive integer. Then

$$T(n)=\Theta(\mathrm{n}^{\log_b^a})+\sum_{j=0}^{\log_b^n-1}a^j f(n/b^j).$$

T(n)=aT(n/b+c)+f(n)? (3)

$$\mathrm{T}(n)=aT(n/b+c)+f(n),\quad n=b^i$$

$$\mathit{find}\ \mathrm{d, such\ that}$$

$$\mathrm{T}(n+d)=aT(n/b+d)+f(n+d),\quad n=b^i$$

$$\mathrm{i}=\log_b^n$$

$$\mathrm{a}^{\log_b^n}=\mathrm{n}^{\log_b^a},$$

$$S(0)=\mathrm{T}(b^0)=\mathrm{T}(1)=C\ \text{(constant)}$$

By definition

$$\mathrm{T}(b^i+d)=aT(b^i/b+d)+f(b^i+d)=aT(b^{i-1}+d)+f(b^i+d)$$

$$let\quad S(i)=T(b^i+d)$$

Changing variables

$$\Rightarrow S(i)=aS(i-1)+f(b^i+d)=a(aS(i-2)+f(b^{i-1}+d))+f(b^i+d)$$

$$=a^2S(i-2)+af(b^{i-1}+d)+f(b^i+d)$$

$$=a^2(aS(i-3)+f(b^{i-2}+d))+af(b^{i-1}+d)+f(b^i+d)$$

$$=a^3S(i-3)+a^2f(b^{i-2}+d)+af(b^{i-1}+d)+f(b^i+d)$$

$$=\ldots$$

$$=a^iS(0)+(a^{i-1}f(b^1+d)+\ldots+af(b^{i-1}+d)+f(b^i+d))$$

$$\Rightarrow \mathrm{T}(n+d)=T(1)(\mathrm{n}^{\log_b^a})+\sum_{j=0}^{\log_b^n-1}a^j f(n/b^j+d)=\Theta(\mathrm{n}^{\log_b^a})+\sum_{j=0}^{\log_b^n-1}a^j f(n/b^j+d).$$

$$\Rightarrow \mathrm{T}(n)=\Theta((\mathrm{n-d})^{\log_b^a})+\sum_{j=0}^{\log_b^n-1}a^j f((n-d)/b^j+d).\ \text{(a, b, d some constant)}$$

T(n)=aT(n/b+c)+f(n)? (4)

- Recall Lemma 4.3!!

$$\Rightarrow T(n) = \Theta((n-d)^{\log_b a}) + \sum_{j=0}^{\log_b n - 1} a^j f((n-d)/b^j + d). \text{ (a, b, d some constant)}$$

$$\text{how about } g(n;d) = \sum_{j=0}^{\log_b n - 1} a^j f((n-d)/b^j + d)?$$

- Lemma 4.3:
  - Let a ≧ 1 and b > 1 be constants, and let f(n) be a nonnegative function defined on exact powers of b. A function g(n) defined over exact powers of by

$$g(n) = \sum_{j=0}^{\log_b n - 1} a^j f(n/b^j)$$

has the following asymptotic bounds for exact powers of b:

$$g(n) = \begin{cases} O(n^{\log_b a}), & \text{if } f(n) = O(n^{\log_b a - \varepsilon}) \text{ for some constant } \varepsilon > 0 \\ \Theta(n^{\log_b a} \lg n), & \text{if } f(n) = \Theta(n^{\log_b a}) \\ \Theta(f(n)), & \text{if } af(n/b) \le cf(n) \text{ for some constant } c < 1 \\ & \text{and for all sufficiently large n} \end{cases}$$

T(n)=aT(n/b+c)+f(n)? (5)

- Can we have a lemma 4.3.b

- Lemma 4.3.b:
  - Let a ≧ 1 and b > 1 be constants, and let f(n) be a nonnegative function defined on exact powers of b (n-d=b^k). A function g(n) defined over exact powers of by

$$g(n;d) = \sum_{j=0}^{\log_b n - 1} a^j f((n-d)/b^j + d)$$

has the following asymptotic bounds for exact powers of b: ??????? (we not prove yet)

$$g(n;d) = \begin{cases} O(n^{\log_b a}), & \text{if } f(n) = O(n^{\log_b a - \varepsilon}) \text{ for some constant } \varepsilon > 0 \\ \Theta(n^{\log_b a} \lg n), & \text{if } f(n) = \Theta(n^{\log_b a}) \\ \Theta(f(n)), & \text{if } af(n/b) \le cf(n) \text{ for some constant } c < 1 \\ & \text{and for all sufficiently large n} \end{cases}$$

Chapter 4

Proof: Lemma 4.3b

given $n-d=b^k$

$$g(n;d)=\sum_{j=0}^{\log_b^n-1} a^j f((n-d)/b^j+d)$$

$$g(n;d)=a^{k-1}f(b^1+d)+\ldots+af(b^{k-1}+d)+f(b^k+d)$$

1. $g(n;d)=O(n^{\log_b^a})$, if $f(n)=O(n^{\log_b^a-\varepsilon})$ for some constant $\varepsilon>0$
2. $g(n;d)=\Theta(n^{\log_b^a}\lg n)$, if $f(n)=\Theta(n^{\log_b^a})$
3. $g(n;d)=\Theta(f(n))$, if $af(n/b+d)\le cf(n+d)$ for some constant $c<1$ and for all sufficiently large n

Proof: Lemma 4.3b(2)

1. $g(n;d)=O(n^{\log_b^a})$, if $f(n)=O(n^{\log_b^a-\varepsilon})$ for some constant $\varepsilon>0$

if $f(n)=O(n^{\log_b^a-\varepsilon})\Rightarrow f(n+d)=O(n^{\log_b^a-\varepsilon})$ if n is sufficiently large

$$\Rightarrow g(n;d)=\sum_{j=0}^{\log_b^n-1} a^j f((n-d)/b^j+d)=O(\sum_{j=0}^{\log_b^n-1} a^j (n/b^j)^{\log_b^a-\varepsilon})$$

$$\sum_{j=0}^{\log_b^n-1} a^j (n/b^j)^{\log_b^a-\varepsilon}=n^{\log_b^a-\varepsilon}(\sum_{j=0}^{\log_b^n-1} a^j (b^{-j})^{\log_b^a-\varepsilon})$$

$$=n^{\log_b^a-\varepsilon}(\sum_{j=0}^{\log_b^n-1} a^j (b^{-j\log_b^a})(b^{j\varepsilon}))=n^{\log_b^a-\varepsilon}(\sum_{j=0}^{\log_b^n-1} a^j (a^{-j})(b^{j\varepsilon}))$$

$$=n^{\log_b^a-\varepsilon}(\sum_{j=0}^{\log_b^n-1} b^{j\varepsilon})=n^{\log_b^a-\varepsilon}(\frac{(b^\varepsilon)^{\log_b^n}-1}{b^\varepsilon-1})=n^{\log_b^a-\varepsilon}(\frac{n^\varepsilon-1}{b^\varepsilon-1})<n^{\log_b^a}$$

given $n-d=b^k$

$$g(n;d)=\sum_{j=0}^{\log_b^n-1} a^j f((n-d)/b^j+d)$$

$$g(n;d)=a^{k-1}f(b^1+d)+\ldots+af(b^{k-1}+d)+f(b^k+d)$$

Proof: Lemma 4.3b(3)

$$2.\ g(n;d) = \Theta(n^{\log_b^a} \lg n), \text{ if } f(n) = \Theta(n^{\log_b^a})$$

if $f(n) = \Theta(n^{\log_b^a}) \Rightarrow f(n+d) = \Theta(n^{\log_b^a})$ if n is sufficiently large

$$\Rightarrow g(n;d) = \sum_{j=0}^{\log_b^n - 1} a^j f((n-d)/b^j + d) = \Theta(\sum_{j=0}^{\log_b^n - 1} a^j (n/b^j)^{\log_b^a})$$

$$\sum_{j=0}^{\log_b^n - 1} a^j (n/b^j)^{\log_b^a} = n^{\log_b^a} (\sum_{j=0}^{\log_b^n - 1} a^j (b^{-j})^{\log_b^a})$$

$$= n^{\log_b^a} (\sum_{j=0}^{\log_b^n - 1} a^j (b^{-j\log_b^a})) = n^{\log_b^a - \varepsilon} (\sum_{j=0}^{\log_b^n - 1} a^j (a^{-j}))$$

$$= n^{\log_b^a} (\sum_{j=0}^{\log_b^n - 1} 1) = (n^{\log_b^a})(\log_b^n)$$

given $n - d = b^k$

$$g(n;d) = \sum_{j=0}^{\log_b^n - 1} a^j f((n-d)/b^j + d)$$

$$g(n;d) = a^{k-1} f(b^1 + d) + \ldots + af(b^{k-1} + d) + f(b^k + d)$$

Proof: Lemma 4.3b(4)

$$3.\ g(n;d) = \Theta(f(\mathrm{n})), \text{ if } af(n/b) \le cf(n) \text{ for some constant } c < 1$$

and for all sufficiently large n

if $af(n/b + d) \le cf(n+d)$

$$\Rightarrow g(n;d) = \sum_{j=0}^{\log_b^n - 1} a^j f((n-d)/b^j + d) \le \sum_{j=0}^{\log_b^n - 1} a^{j-1}(cf((n-d)/b^{j-1} + d))$$

$$\le \sum_{j=0}^{\log_b^n - 1} a^{j-2}(c^2 f((n-d)/b^{j-2} + d))$$

....

$$\le \sum_{j=0}^{\log_b^n - 1} a^{j-\mathrm{j}}(c^{\mathrm{j}} f((n-d)/b^{j-\mathrm{j}} + d)) \le \sum_{j=0}^{\log_b^n - 1} c^{\mathrm{j}} f((n-d)+d) = f(n)\sum_{j=0}^{\log_b^n - 1} c^{\mathrm{j}} \le f(n)\sum_{j=0}^{\infty} c^{\mathrm{j}}$$

$$\le f(n)(\frac{1}{1-c}) \text{ (since } c < 1)$$

$$= O(f(n))$$

given $n - d = b^k$

$$g(n;d) = \sum_{j=0}^{\log_b^n - 1} a^j f((n-d)/b^j + d)$$

$$g(n;d) = a^{k-1} f(b^1 + d) + \ldots + af(b^{k-1} + d) + f(b^k + d)$$

Drill

- Problems 4-1

$$a. T(n) = 2T(n/2) + n^4.$$
$$b. T(n) = T(7n/10) + n.$$
$$c. T(n) = 16T(n/4) + n^2.$$
$$d. T(n) = 7T(n/3) + n^2.$$
$$e. T(n) = 7T(n/2) + n^2.$$
$$f. T(n) = 2T(n/4) + n^{1/2}.$$
$$g. T(n) = T(n-2) + n^2.$$

T(n)=4T(n/2+2)+n?

- Cannot apply master method!!

Do transformation!
We would like to find
From T(n)=aT(n/b+d)+f(n) ➔ T(n+c)=aT(n/b+c)+f(n+c)

(n+c)/b+d=n/b+c➔ c/b+d=c➔c=d/(1-1/b)
e.g. b=d=2, ➔ c=2/(1-1/2)=4

$$T(n) = \mathrm{a}T(\frac{n}{2}+2) + f(n)$$
$$\Rightarrow T(n+4) = \mathrm{a}T(\frac{n}{2}+4) + f(n+4)$$

Instead of find the answer for T(n), we try to solve T(n+4)
Once we find T(n+4), we change the variable back

T(n)=n^(1/2)T(n^(1/2))+n

- Cannot apply master method!!

$$T(n)=\sqrt{\mathrm{n}}\,T(\sqrt{\mathrm{n}})+n$$
$$=\mathrm{n}^{1/2}T(\mathrm{n}^{1/2})+n=\mathrm{n}^{1/2}(\mathrm{n}^{1/4}T(\mathrm{n}^{1/4})+\mathrm{n}^{1/2})+n$$
$$=\mathrm{n}^{1/2+1/4}T(\mathrm{n}^{1/4})+\mathrm{n}+n=\mathrm{n}^{1/2+1/4}(\mathrm{n}^{1/8}T(\mathrm{n}^{1/8})+\mathrm{n}^{1/4})+\mathrm{n}+n$$
$$=\mathrm{n}^{1/2+1/4+1/8}T(\mathrm{n}^{1/8})+n+(n+n)$$
$$=\ldots$$
$$=\mathrm{n}^{1/2+1/4+1/8+\ldots+1/2^k}T(\mathrm{n}^{1/2^k})+kn=\mathrm{n}^{1-1/2^k}T(\mathrm{n}^{1/2^k})+kn$$

T(n)=T(n/2)+T(n/4)+T(n/8)+n

- Cannot apply master method!!

```
0  2  4   8  | 0
   2  4   8  16  | 2
      4   8  16  32  | 4
          8  16  32  64  | 8
             16  32  64  128 | 16
                 32  64  128  256  | 32
                   ...............
```

$$\mathrm{F}(0)=\mathrm{F}(1)=1,\ F(2)=2$$
$$F(i)=F(i-1)+F(i-2)+F(i-3), i>2$$

$$\mathrm{T}(n)=T(n/2)+T(n/4)+T(n/8)+n$$
$$=(1+1)T(n/4)+(1+1)T(n/8))+\mathrm{T}(n/16)+n+n/2$$
$$=(1+1+2)T(n/8)+(1+2)T(n/16)+2T(n/32)+n+n/2+2n/4$$
$$=(1+2+4)T(n/16)+(2+4)T(n/32)+4T(n/64)+n+n/2+2n/4+4n/8$$
$$=(2+4+7)T(n/32)+(4+7)T(n/64)+7\mathrm{T}(n/128)+\ldots$$
$$=(4+7+13)T(n/64)+(7+13)\mathrm{T}(n/128)+13\mathrm{T}(n/256)+\ldots$$
$$=(7+13+24)T(n/64)+(13+24)\mathrm{T}(n/256)+24\mathrm{T}(n/512)+\ldots$$

Problem 4.4: Fibonacci numbers

- Generating function

$$F(z)=\sum_{i=0}^{\infty}F_i z^i=0+z+z^2+2z^3+3z^4+5z^5+8z^6+\ldots$$

$$F_i=\begin{cases}0, & \mathrm{i}=0\\ 1, & \mathrm{i}=1\\ F_{i-1}+F_{i-2}, & \mathrm{i}>1\end{cases}$$

a. show that: $F(z)=z+zF(z)+z^2F(z)$

b. show that: $F(z)=\dfrac{z}{1-z-z^2}=\dfrac{z}{(1-\phi z)(1-\bar{\phi}z)}$

$$=\frac{1}{\sqrt{5}}\left(\frac{1}{1-\phi z}-\frac{1}{1-\bar{\phi}z}\right)$$

where

$$\phi=\frac{1+\sqrt{5}}{2}=1.61803\ldots$$

$$\bar{\phi}=\frac{1-\sqrt{5}}{2}=-0.61803\ldots$$

c. show that: $F(z)=\sum_{i=0}^{\infty}\dfrac{1}{\sqrt{5}}(\phi^i-\bar{\phi}^i)z^i$

d. use part (c) to prove that $F_i=\phi^i/\sqrt{5}, i>0$, rounded to the nearest integer. (hint: observe that $|\bar{\phi}|<1$)

Problem 4.4: Fibonacci numbers(2)

$$F(z)=\sum_{i=0}^{\infty}F_i z^i=0+z+z^2+2z^3+3z^4+5z^5+8z^6+\ldots$$

$$F_i=\begin{cases}0, & \mathrm{i}=0\\ 1, & \mathrm{i}=1\\ F_{i-1}+F_{i-2}, & \mathrm{i}>1\end{cases}$$

a. show that: $F(z)=z+zF(z)+z^2F(z)$

$$z+zF(z)+z^2F(z)=z+z\sum_{i=0}^{\infty}F_iz^i+z^2\sum_{i=0}^{\infty}F_iz^i$$

$$=z+\sum_{i=0}^{\infty}F_iz^{i+1}+\sum_{i=0}^{\infty}F_iz^{i+2}=z+\sum_{i=1}^{\infty}F_{i-1}z^i+\sum_{i=2}^{\infty}F_{i-2}z^i$$

$$=z+\sum_{i=2}^{\infty}F_{i-1}z^i+\sum_{i=2}^{\infty}F_{i-2}z^i=z+\sum_{i=2}^{\infty}(F_{i-1}+F_{i-2})z^i$$

$$=z+\sum_{i=2}^{\infty}F_iz^i=\sum_{i=1}^{\infty}F_iz^i=\sum_{i=0}^{\infty}F_iz^i$$

Problem 4.4: Fibonacci numbers(3)

$$F(z)=\sum_{i=0}^{\infty} F_i z^i = 0+z+z^2+2z^3+3z^4+5z^5+8z^6+...$$

$$F_i=\begin{cases}0, & \text{i}=0\\ 1, & \text{i}=1\\ F_{i-1}+F_{i-2}, & \text{i}>1\end{cases}$$

$$\text{b. show that}: F(z)=\frac{z}{1-z-z^2}=\frac{z}{(1-\phi z)(1-\bar{\phi} z)}$$

$$=\frac{1}{\sqrt{5}}\left(\frac{1}{1-\phi z}-\frac{1}{1-\bar{\phi} z}\right)$$

where

$$\phi=\frac{1+\sqrt{5}}{2}=1.61803...$$

$$\bar{\phi}=\frac{1-\sqrt{5}}{2}=-0.61803...$$

proof : from (a),

we have $F(z)=z+zF(z)+z^2F(z)$

$$\Rightarrow F(z) - zF(z) - z^2F(z)=z$$

$$\Rightarrow F(z)=\frac{z}{1-z-z^2}=\frac{z}{(1-\phi z)(1-\bar{\phi} z)}$$

such that : $(\phi+\bar{\phi})=1; \phi\bar{\phi}=-1$

$(\phi,\bar{\phi})$ are the roots of equation : $1 - \text{x} - \text{x}^2$

$$\Rightarrow \phi=\frac{1+\sqrt{5}}{2}=1.61803..., \text{and } \bar{\phi}=\frac{1-\sqrt{5}}{2}=-0.61803...$$

Problem 4.4: Fibonacci numbers(4)

$$F(z)=\sum_{i=0}^{\infty} F_i z^i = 0+z+z^2+2z^3+3z^4+5z^5+8z^6+...$$

$$F_i=\begin{cases}0, & \text{i}=0\\ 1, & \text{i}=1\\ F_{i-1}+F_{i-2}, & \text{i}>1\end{cases}$$

$$\text{c. show that}: F(z)=\sum_{i=0}^{\infty}\frac{1}{\sqrt{5}}(\phi^i-\bar{\phi}^i)z^i$$

proof : from (b), $F(z)=\frac{1}{\sqrt{5}}\left(\frac{1}{1-\phi z}-\frac{1}{1-\bar{\phi} z}\right)$

recall $\frac{1}{1-r}=\sum_{i=0}^{\infty} r^i, \quad |\text{r}|<1$

$$\frac{1}{1-\phi z}=\sum_{i=0}^{\infty}(\phi z)^i, \quad \text{assume } |\phi z|<1,$$

$$\frac{1}{1-\bar{\phi} z}=\sum_{i=0}^{\infty}(\bar{\phi} z)^i, \quad \text{assume } |\bar{\phi} z|<1,$$

$$F(z)=\frac{1}{\sqrt{5}}\left(\frac{1}{1-\phi z}-\frac{1}{1-\bar{\phi} z}\right)=\frac{1}{\sqrt{5}}\left(\sum_{i=0}^{\infty}(\phi z)^i-(\bar{\phi} z)^i\right)$$

$$=\sum_{i=0}^{\infty}\frac{1}{\sqrt{5}}(\phi^i-\bar{\phi}^i)z^i \qquad (, \text{ assume } |\bar{\phi} z|<1 \text{ and } |\phi z|<1)$$

$(note: given\, \phi,\bar{\phi}$ such z exists)

Problem 4.4: Fibonacci numbers(5)

$$F(z)=\sum_{i=0}^{\infty}F_i z^i = 0+z+z^2+2z^3+3z^4+5z^5+8z^6+\ldots$$

$$F_i=\begin{cases}0, & \text{i}=0\\ 1, & \text{i}=1\\ F_{i-1}+F_{i-2}, & \text{i}>1\end{cases}$$

d. use part (c) to prove that $F_i=\phi^i/\sqrt{5}, i>0,$ rounded to the nearest integer.(hint : observe that $|\bar{\phi}|<1$)

proof : from (c), show that : $F(z)=\sum_{i=0}^{\infty}\frac{1}{\sqrt{5}}(\phi^i-\bar{\phi}^i)z^i$

$$\Rightarrow F_i=\frac{1}{\sqrt{5}}(\phi^i-\bar{\phi}^i)$$

$=\frac{1}{\sqrt{5}}\phi^i$ rounded to the nearest integer.(since $|\bar{\phi}|<1$)

Appendix:

Review: Asymptotic notation

3.1 Asymptotic notation

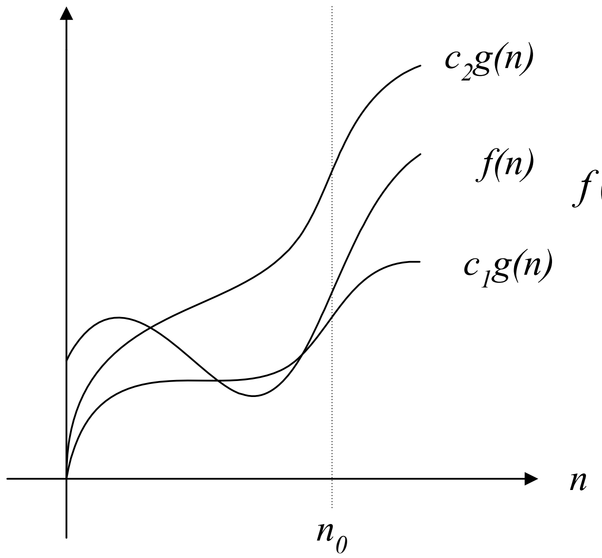
$$\Theta(g(n)) = \{f(n) \mid \exists c_1, c_2, n_0 \text{ s.t. } 0 \le c_1 g(n) \le f(n) \le c_2 g(n)$$
$$\text{for all } n \ge n_0\}$$

$$f(n) = \Theta(g(n))$$

$\Rightarrow$ *g*(*n*) is an asymptotic tight bound for *f*(*n*).

- ``=’’ abuse

- The definition of required every member of be asymptotically nonnegative.



$f(n) = \Theta(g(n))$

Example:

$$\frac{n^2}{14} \le \frac{n^2}{2} - 3n \le \frac{n^2}{2} \; if \; n>7.$$

$$6n^3 \ne \Theta(n^2)$$

$f(n) = an^2 + bn + c$, a, b, c constants, $a>0$.

$\Rightarrow f(n)=\Theta(n^2)$.

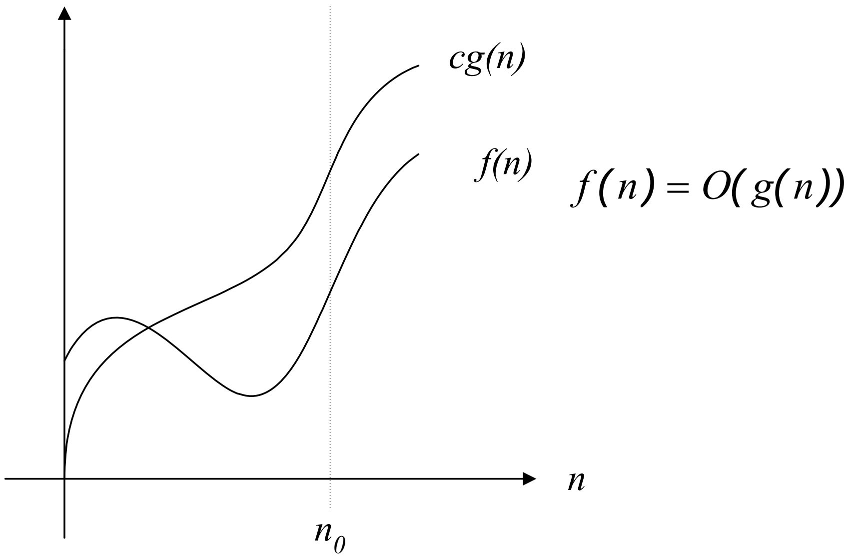
- In general,

$p(n) = \sum_{i=0}^{d} a_i n^i$ where a_i are constant with $a_d > 0$.

Then $P(n) = \Theta(n^d)$.

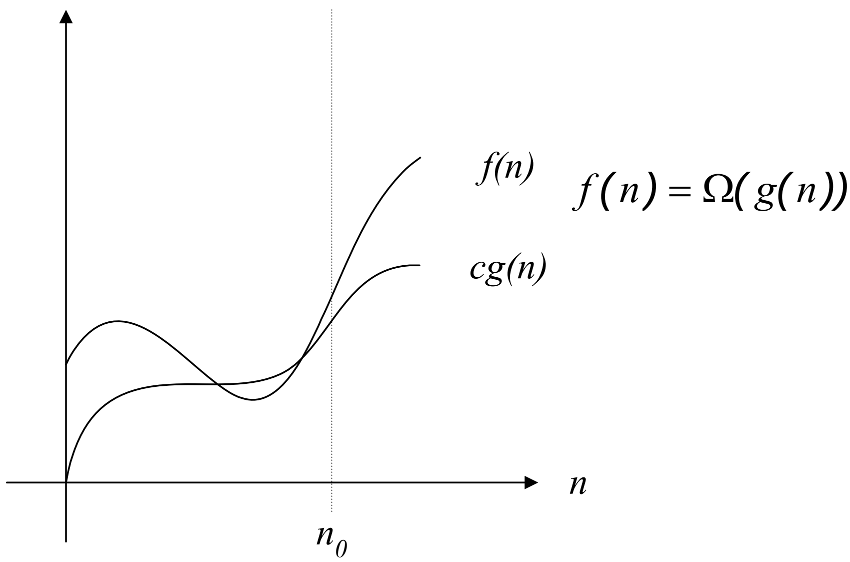
asymptotic upper bound

$$O(g(n)) = \{ f(n) \mid \exists c, n_0 \text{ s.t. } 0 \le f(n) \le cg(n)\ \forall n \ge n_0 \}$$



asymptotic lower bound

$$\Omega(g(n)) = \{ f(n) \mid \exists c, n_0 \text{ s.t. } 0 \le cg(n) \le f(n)\ \forall n \ge n_0 \}$$



Theorem 3.1.

- For any two functions *f*(*n*) and *g*(*n*), $f(n) = \Theta(g(n))$ if and only if $f(n) = O(g(n))$ and $f(n) = \Omega(g(n))$.

- $$o(g(n)) = \{ f(n) | \forall c, \exists n_0 \ \forall n > n_0, 0 \leq f(n) \cancel{\leq} cg(n) \}$$
- $$f(n) = o(g(n)) \Leftrightarrow \lim_{n \to \infty} \frac{f(n)}{g(n)} = 0$$
- $$\omega(g(n)) = \{ f(n) | \forall c, \exists n_0 \ \forall n > n_0, 0 \leq cg(n) \cancel{\leq} f(n) \}$$
- $$f(n) = \omega(g(n)) \Leftrightarrow \lim_{n \to \infty} \frac{f(n)}{g(n)} = \infty$$