Soluzione delle relazioni di ricorrenza quando n non è potenza di β

- Consideriamo la seguente relazione di ricorrenza:

$$T(n) \leq \begin{cases} c_0 & \text{se } n \leq c \\ \alpha T(n/\beta) + f(n) & \text{altrimenti} \end{cases}$$

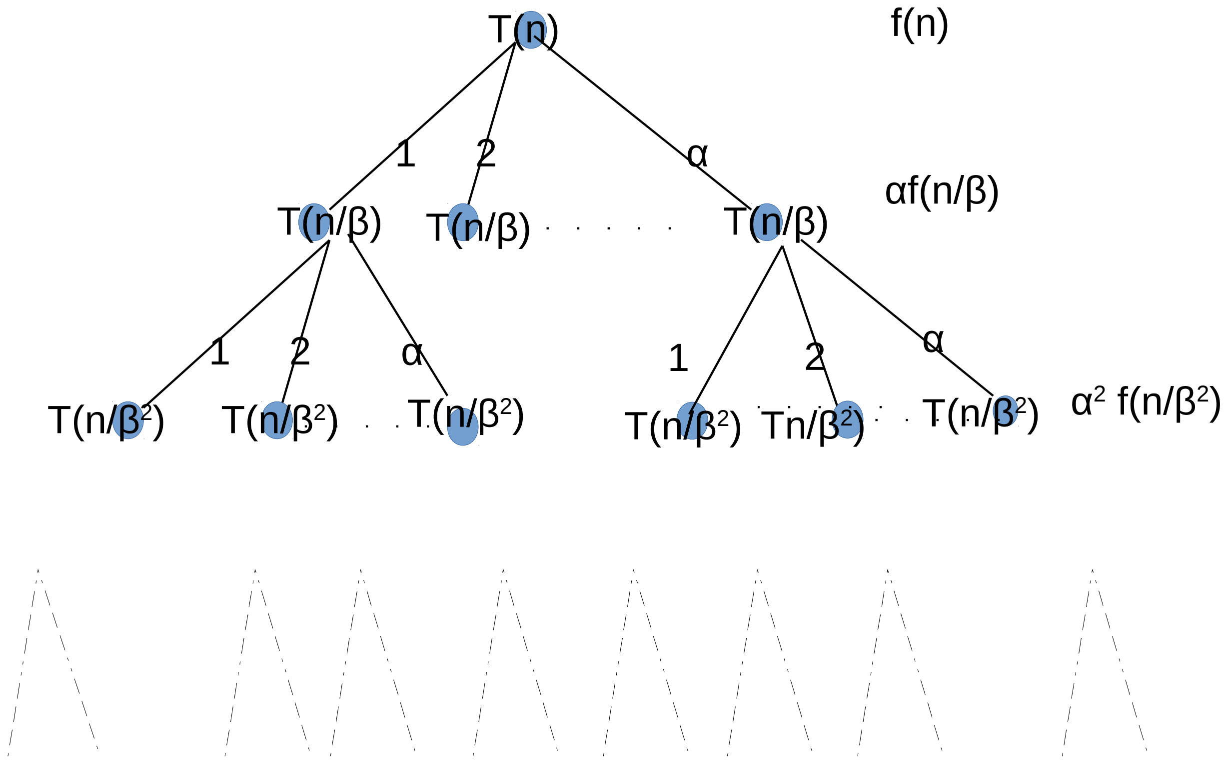
  con $\alpha \geq 1$ e $\beta > 1$ costanti.
- Quando n non è una potenza di β la taglia di ciascun sottoproblema è $\lceil n/\beta \rceil$ oppure $\lfloor n/\beta \rfloor$.
- Siccome vogliamo stabilire un limite superiore per $T(n)$ mettiamoci nel caso peggiore in cui la taglia di ciascun sottoproblema è $\lceil n/\beta \rceil$ Consideriamo quindi la seguente relazione di ricorrenza:

$$T(n) \leq \begin{cases} c_0 & \text{se } n \leq c \\ \alpha T(\lceil n/\beta \rceil) + f(n) & \text{altrimenti} \end{cases}$$

  con $\alpha \geq 1$ e $\beta > 1$ costanti. Assumiamo questa volta anche che $c \geq 2\beta/(\beta - 1)$. Ai fini pratici non cambia niente perchè un algoritmo il cui tempo di esecuzione dipende dalla dimensione dell'input richiede tempo costante per input la cui dimensione è limitata da una qualsiasi costante.

Soluzione delle relazioni di ricorrenza quando n non è potenza di β

Nell'albero delle ricorsioni che abbiamo utilizzato per il calcolo di $T(n)$ quando n è una potenza di β dobbiamo sostituire ciascun n/β sul livello 1 con $N_1 = \lceil n/\beta \rceil$, ciascun n/β^2 sul livello 2 con $N_2 = \lceil \lceil n/\beta \rceil / \beta \rceil$, ciascun n/β^3 sul livello 3 con $N_3 = \lceil \lceil \lceil n/\beta \rceil / \beta \rceil \beta \rceil$, e così via.



Sia h l'altezza dell'albero (h+1 livelli). Per i<h, Il tempo di esecuzione per tutte le chiamate ricorsive a livello i e` al piu` $\alpha^i f(n/\beta^i)$.

Soluzione delle relazioni di ricorrenza quando n non è potenza di β

- Osserviamo che

$$
\begin{aligned}
N_i &= \lceil N_{i-1}/\beta \rceil < N_{i-1}/\beta + 1 \\
&= \lceil N_{i-2}/\beta \rceil/\beta + 1 < N_{i-2}/\beta^2 + 1/\beta + 1 \\
&= \lceil N_{i-3}/\beta \rceil/\beta^2 + 1/\beta + 1 < N_{i-3}/\beta^3 + 1/\beta^2 + 1/\beta + 1 \\
&= \ldots \\
&= \lceil N_1/\beta \rceil/\beta^{i-2} + 1/\beta^{i-3} + 1/\beta^{i-4} + \ldots + 1 \\
&< N_1/\beta^{i-1} + 1/\beta^{i-2} + 1/\beta^{i-3} + 1/\beta^{i-4} + \ldots + 1 \\
&= \lceil n/\beta \rceil/\beta^{i-1} + 1/\beta^{i-2} + 1/\beta^{i-3} + 1/\beta^{i-4} + \ldots + 1 \\
&< n/\beta^i + 1/\beta^{i-1} + 1/\beta^{i-2} + 1/\beta^{i-3} + \ldots + 1 \\
&= n/\beta^i + \sum_{j=0}^{i-1} (1/\beta)^j < n/\beta^i + \sum_{j=0}^{\infty} (1/\beta)^j \\
&= n/\beta^i + 1/(1-1/\beta) = n/\beta^i + \beta/(\beta-1)
\end{aligned}
$$

Soluzione delle relazioni di ricorrenza quando n non è potenza di β

- N_i è la dimensione dell'input delle chiamate ricorsive a livello i. L'algoritmo non effettua chiamate ricorsive quando $N_i \leq c$. Quindi la ricorrenza non sarà più applicata quando si arriva al livello i per cui per la prima volta $N_i \leq c$. Questo livello i sarà quindi l'ultimo livello dell'albero.
- Siccome abbiamo visto che $N_i < n/\beta^i + \beta/(\beta-1)$, basta imporre $n/\beta^i + \beta/(\beta-1) \leq c$ che è soddisfatta per $i \geq \left\lceil \log_\beta \frac{n}{c-\beta/(\beta-1)} \right\rceil$. L'ultimo livello dell'albero è quindi il livello $i = \left\lceil \log_\beta \frac{n}{c-\beta/(\beta-1)} \right\rceil$.
- Nel seguito poniamo $c_1 = c - \beta/(\beta-1)$. Si noti che siccome $c \geq 2\beta/(\beta-1)$ allora $c_1 \geq 2\beta/(\beta-1)$.
- Ciascun nodo sul livello $\lceil \log_\beta n/c_1 \rceil$ corrisponde al tempo $T(N_{\lceil \log_\beta n/c_1 \rceil}) \leq T(c) \leq c_0$. Il tempo totale per eseguire le $\alpha^{\lceil \log_\beta n/c_1 \rceil}$ chiamate ricorsive in quest'ultimo livello è quindi $\leq \alpha^{\lceil \log_\beta n/c_1 \rceil} c_0$.
- Per $i < \lceil \log_\beta n/c_1 \rceil$, il tempo per eseguire tutte le chiamate sul livello i è $\alpha^i f(N_i)$.
- Sommando su tutti i livelli si ha

$$T(n) \leq \alpha^{\lceil \log_\beta n/c_1 \rceil} c_0 + \sum_{i=0}^{\lceil \log_\beta n/c_1 \rceil - 1} \alpha^i f(N_i). \tag{3}$$

Soluzione delle relazioni di ricorrenza quando n non è potenza di β

- Come abbiamo fatto per il caso in cui n è potenza di β, vogliamo stimare $T(n)$ quando la funzione $f(n)$ nella relazione di ricorrenza è limitata da $c'n^k$, dove c' e k sono due costanti tali che $k \geq 0$, $c' > 0$.
- Dalla (3) nella slide precedente si ha che

$$T(n) \leq \alpha^{\lceil \log_\beta n/c_1 \rceil} c_0 + \sum_{i=0}^{\lceil \log_\beta n/c_1 \rceil - 1} \alpha^i c'(N_i)^k$$

Siccome $N_i < n/\beta^i + \beta/(\beta-1)$, si ha

$$T(n) < \alpha^{\lceil \log_\beta n/c_1 \rceil} c_0 + \sum_{i=0}^{\lceil \log_\beta n/c_1 \rceil - 1} \alpha^i c'(n/\beta^i + \beta/(\beta-1))^k. \qquad (4)$$

Per ogni i della sommatoria si ha $\beta^i \leq \beta^{\lceil \log_\beta n/c_1 \rceil - 1} < \beta^{\log_\beta n/c_1} = n/c_1$ per cui $n/\beta^i > c_1 \geq \beta/(\beta-1)$. Questa diseguaglianza insieme alla (4) implica

$$\begin{aligned} T(n) &< \alpha^{\lceil \log_\beta n/c_1 \rceil} c_0 + \sum_{i=0}^{\lceil \log_\beta n/c_1 \rceil - 1} \alpha^i c'(2n/\beta^i)^k \\ &= \alpha^{\lceil \log_\beta n/c_1 \rceil} c_0 + 2^k c' n^k \sum_{i=0}^{\lceil \log_\beta n/c_1 \rceil - 1} (\alpha/\beta^k)^i \end{aligned}$$

SOLUZIONE DELLE RELAZIONI DI RICORRENZA QUANDO n NON È POTENZA DI β

- Abbiamo visto che se $f(n) \leq c' n^k$, con $c' > 0$ e $k \geq 0$ costanti, allora

$$T(n) < \alpha^{\lceil \log_\beta n/c_1 \rceil} c_0 + 2^k c' n^k \sum_{i=0}^{\lceil \log_\beta n/c_1 \rceil - 1} (\alpha/\beta^k)^i.$$

- Poniamo $c_1' = 2^k c'$ in questa disequazione (c_1' è una costante maggiore di 0 perché $k \geq 0$ e $c' > 0$ ed entrambe sono costanti). Otteniamo quindi

$$T(n) < \alpha^{\lceil \log_\beta n/c_1 \rceil} c_0 + c_1' n^k \sum_{i=0}^{\lceil \log_\beta n/c_1 \rceil - 1} (\alpha/\beta^k)^i. \qquad (5)$$

- per il caso in cui n è potenza di β, abbiamo dimostrato che

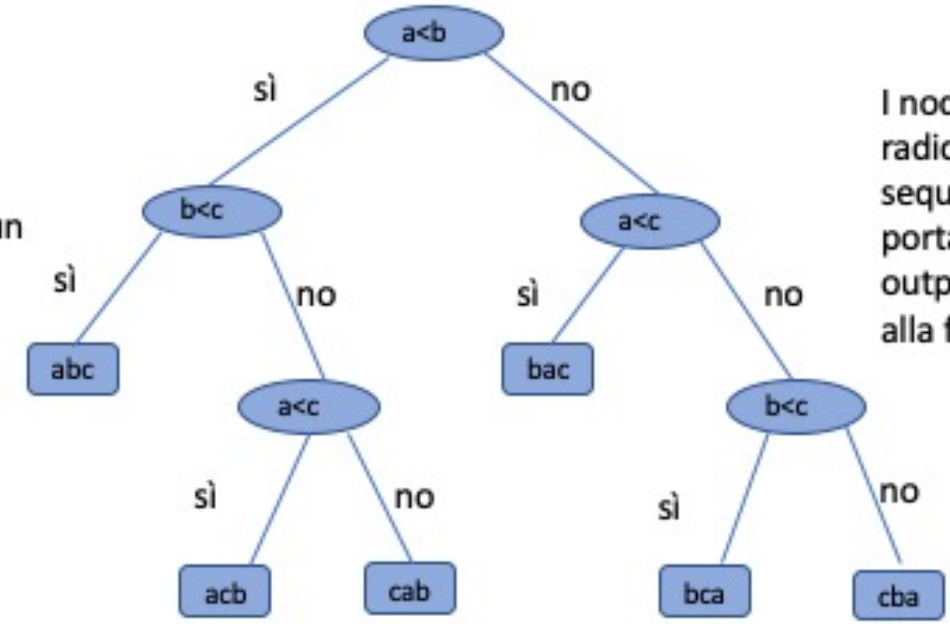
$$T(n) \leq \alpha^{\lceil \log_\beta n/c \rceil} c_0 + c' n^k \sum_{i=0}^{\lceil \log_\beta n/c \rceil - 1} (\alpha/\beta^k)^i.$$

  e usato questa disuguaglianza per ottenere una stima asintotica di $T(n)$.
- Questa disuguaglianza è identica alla (5) se sostituiamo c con c_1 e c' con c_1' (c_1 è una costante ≥ 0 come c e c_1' è una costante > 0 come c').
- Possiamo quindi proseguire nella stima di $T(n)$ nei tre casi $\alpha = \beta^k$, $\alpha < \beta^k$, $\alpha > \beta^k$ così come abbiamo fatto per il caso in cui n è potenza di β e riottenere le stesse stime asintotiche.

Albero di decisione di un particolare algoritmo basato sui confronti per ordinare 3 numeri

I nodi interni sono associati ai confronti

Ciascuna foglia e` associata ad un possible ordinamento (permutazione)



I nodi lungo un percorso dalla radice ad una foglia forniscono la sequenza di confronti che hanno portato l'algoritmo a dare in output l'ordinamento associato alla foglia

Altezza dell'albero = massimo numero di confronti fatti in un'esecuzione dall'algoritmo

Limite inferiore per gli algoritmi di ordinamento basati su confronti

- L'albero di decisione di un qualsiasi algoritmo di ordinamento deve avere una foglia per ogni possibile ordinamento dell'input
- Se l'input consiste di n numeri allora l'albero di decisione deve contenere almeno n! foglie. Sia h l'altezza dell'albero.
- Il massimo numero di foglie in un albero binario di altezza h e` 2^h
- Ne consegue che l'altezza h dell'albero deve essere tale che

  $2^h \geq n! \rightarrow h \geq \log n! \geq \log (n/2)^{n/2} = n/2 \log(n) - n/2$

  $\rightarrow h = \Omega(n\log n)$

- Siccome h=numero confronti fatti nel caso pessimo dall'algoritmo allora abbiamo dimostrato che il numero di confronti effettuati nel caso pessimo di da qualsiasi algoritmo di ordinamento basato su confronti e` $\Omega(n\log n)$.