Ricapitolando

MERGESORT $(A, i, j)$

IF $i < j$ THEN
$q = \frac{i+j}{2}$

MERGESORT $(A, i, q)$
MERGESORT $(A, q+1, r)$
MERGE $(A, i, q, r)$

$n$ = numero di elementi nella sottosequenza tra $i$ e $j$

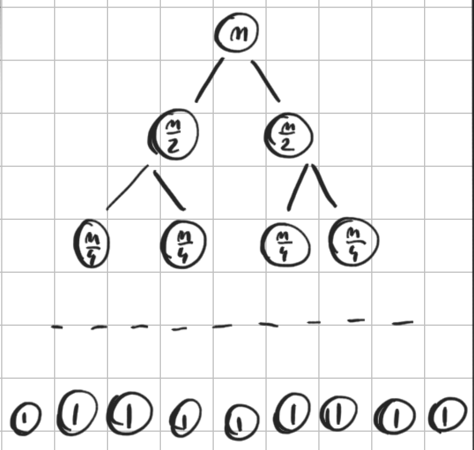
$n = j - i + 1$

$$T(n) = \begin{cases} \Theta 1 & n \leq 1 \quad \text{(caso base)} \\ 2 \cdot T\left(\frac{n}{2}\right) + \Theta(n) & n > 1 \end{cases}$$

## ANALISI DI ALGORITMI RICORSIVI

Analizziamo il tempo d'esecuzione del mergesort
A differenza del fattoriale qui ogni chiamata ha 2 chiamate ricorsive generando un albero BINARIO (albero di grado 2 ovvero in cui ogni livello genera al più 2 figli)

| | LIVELLI | ALBERO | T(n) (per nodo) |
|---|---|---|---|
| radice | 0 | | $\Theta n$ |
| | 1 | | $\Theta \frac{n}{2}$ |
| | 2 | | $\Theta \frac{n}{4}$ |
| | ... | | |
| foglie | h | | $\Theta 1$ |




La radice (primo nodo) prende in input n dati e apporta di suo un tempo $\Theta n$. Genera 2 chiamate figlie il cui tempo di esecuzione è proporzionale alla dimensione dell'input $\left(\frac{n}{2}\right)$ e così via.

Il tempo locale (di ogni chiamata) e' dato dalla somma dei tempi di ogni livello dell'albero.

Si ricordi che per le foglie (caso base) abbiamo $\Theta 1$ e che si trovano al livello h (ALTEZZA dell'albero)

$$T(n) = n° \text{ foglie} + \sum_{0}^{h-1} \Theta n$$

Il livello i-esimo di un albero binario pieno (che per ogni livello genera il numero massimo di nodi) possiede $2^i$ nodi.

Come troviamo l'altezza h? (livello delle foglie)

Il livello i-esimo avrà in input $\frac{n}{2^i}$.

Quando $\frac{n}{2^i}=1$ avremo il livello delle foglie (caso base) e quindi l'altezza $h$.

$$\frac{n}{2^i}=1 \rightarrow n=2^i \rightarrow \log_2 n=\log_2(2^i) \rightarrow \log_2 n=i$$

Quindi $h=\log_2 n$. Possiamo sostituire per avere:

$$T(n)=2^{\log_2 n}+\sum_{i=0}^{\log_2 n-1} n = n+\sum_{i=0}^{\log_2 n-1} n = \sum_{i=0}^{\log_2 n} n = n(\log_2 n+1)=\Theta(n\log n)$$

## CONFRONTO MERGE E INSERTION SORT

Ci saranno casi in cui l'insertion sort sarà sicuramente migliore ma vediamo la crescita:

l'insertion sort cresce con un tempo $\Theta\, n^2$ mentre il merge con $\Theta\, n\log n$

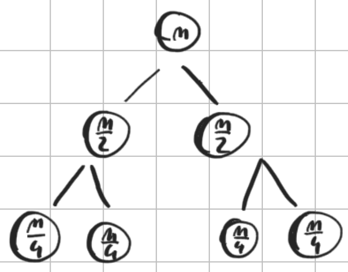
È banale notare che $\frac{n^{\cancel{2}}}{\cancel{n}\log_2 n}$ cresce esponenzialmente, quindi l'insertion sort va esponenzialmente peggio.

## OSS

In linea teorica il merge sort ha un tempo di esecuzione ottimale nel suo caso medio e peggiore

ES. (equazioni di ricorrenza)

$$T(n) = \begin{cases} \Theta 1 & n \le 2 \\ 2T\left(\frac{n}{2}\right) + \Theta n^2 & n > 2 \end{cases}$$

| LIVELLO | ALBERO | Tempo locale per livello |
|---|---|---|
| 0 |  | $1 \cdot (n^2)$ |
| 1 | | $+ \quad 2 \cdot \left(\frac{n}{2}\right)^2$ |
| 2 | | $+ \quad 4 \cdot \left(\frac{n}{4}\right)^2$ |
| --- | ---- | |
| h | | |

Ogni livello ha $2^i$ nodi quindi il tempo locale è:

$$\underbrace{2^i}_{\text{n° nodi}} \underbrace{\left(\frac{n}{2^i}\right)^2}_{\text{costo x nodo}} = \cancel{2^i} \frac{n^2}{(2^i)^{\cancel{2}}} = \frac{n^2}{2^i}$$

Il livello i delle foglie è (caso base)

$$\frac{n}{2^i} = 2 \to n = 2 \cdot 2^i \to \log_2 n = \log_2 2^{i+1} \to i = \log_2 n - 1 = h$$

Calcoliamo la sommatoria

$$T(n) = \underbrace{2^h}_{\text{n° di foglie}} + \underbrace{\sum_{i=0}^{h-1}}_{\text{n° di livelli}} \underbrace{\left(\frac{n^2}{2^i}\right)}_{\text{tempo locale}} = 2^{\log_2 n - 1} + \sum_{i=0}^{\log_2 n - 2} \left(\frac{n^2}{2^i}\right)$$

$$2^{\log_2 n-1} = 2^{\log_2 n} \cdot \frac{1}{2} = \frac{n}{2}$$ sostituisco

$$\frac{n}{2} + n^2 \sum_{i=0}^{\log_2 n-2} \left(\frac{1}{2}\right)^i =$$

La seconda parte e' una forma particolare di serie geometrica

Ricordiamo che

$$\sum_{i=0}^{k} x^i = \frac{x^{k+1}-1}{x-1}$$ per $x \geq 1$ e DIVERGE

$$\sum_{i=0}^{k} x^i = \frac{1}{1-x}$$ per $0<x<1$ e CONVERGE a 0

Nel nostro caso $x=\frac{1}{2}$ e quindi il limite della sommatoria $\rightarrow 0$

Sappiamo quindi che vale sempre

$$1 = \sum_{i=0}^{0} x^i \leq \sum_{i=0}^{k} x^i \leq \sum_{i=0}^{\infty} x^i = \frac{1}{1-x}$$

0 elementi      tutti gli elementi

Ma quindi $\sum_{i=0}^{k} x^i$ (che e' il nostro caso) e' compreso tra due costanti

(con $x=\frac{1}{2}$) e quindi $\sum_{0}^{\log_2 n-2} \left(\frac{1}{2}\right)^i = \Theta 1$. In definitiva:

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