

Università degli Studi di Trieste

Corsi di ingegneria industriale e navale

Esercitazioni di Fondamenti di Informatica

Giacomo Strangolino

mailto/chat: delleceste@gmail.com

Materiale didattico anche su:

<http://www.giacomos.it>

(<http://www.giacomos.it/didattica/units/2011/>)

Lezione 6 (30/11/2011)

Funzioni ricorsive

- Una funzione può chiamare se stessa. Una funzione si dice ricorsiva se un'istruzione al suo interno chiama la funzione stessa. La ricorsione è un processo con il quale si definisce qualcosa in termini di se stesso (definizione circolare)

Lezione 6

Funzioni ricorsive: calcolo del fattoriale

- Il calcolo del *fattoriale* di un intero n è dato dal prodotto di tutti i numeri tra uno e n
- Quando la funzione ricorsiva per il calcolo del fattoriale è chiamata con argomento pari ad 1, allora essa restituisce 1. Altrimenti, restituisce il valore di $fattoriale(n - 1) * n$
- Es: $fattoriale(2)$. La prima chiamata causa una seconda chiamata ricorsiva con argomento 1. Quest'ultima restituisce quindi 1, che poi viene moltiplicato per 2 (il valore originale della chiamata).

Lezione 6

Funzioni ricorsive: calcolo del fattoriale

- Provate a tenere traccia delle chiamate per n che vale 3 e così via...
- Un buon metodo per verificare a che punto della computazione ricorsiva ci si trova è quello di inserire delle *print* nella funzione atte a evidenziare ad esempio il risultato parziale e il livello.

Lezione 6

Funzioni ricorsive

- Riscrivere la funzione fattoriale in modo ricorsivo.
- Scrivere un'implementazione in *python* dell'algoritmo *merge sort*.

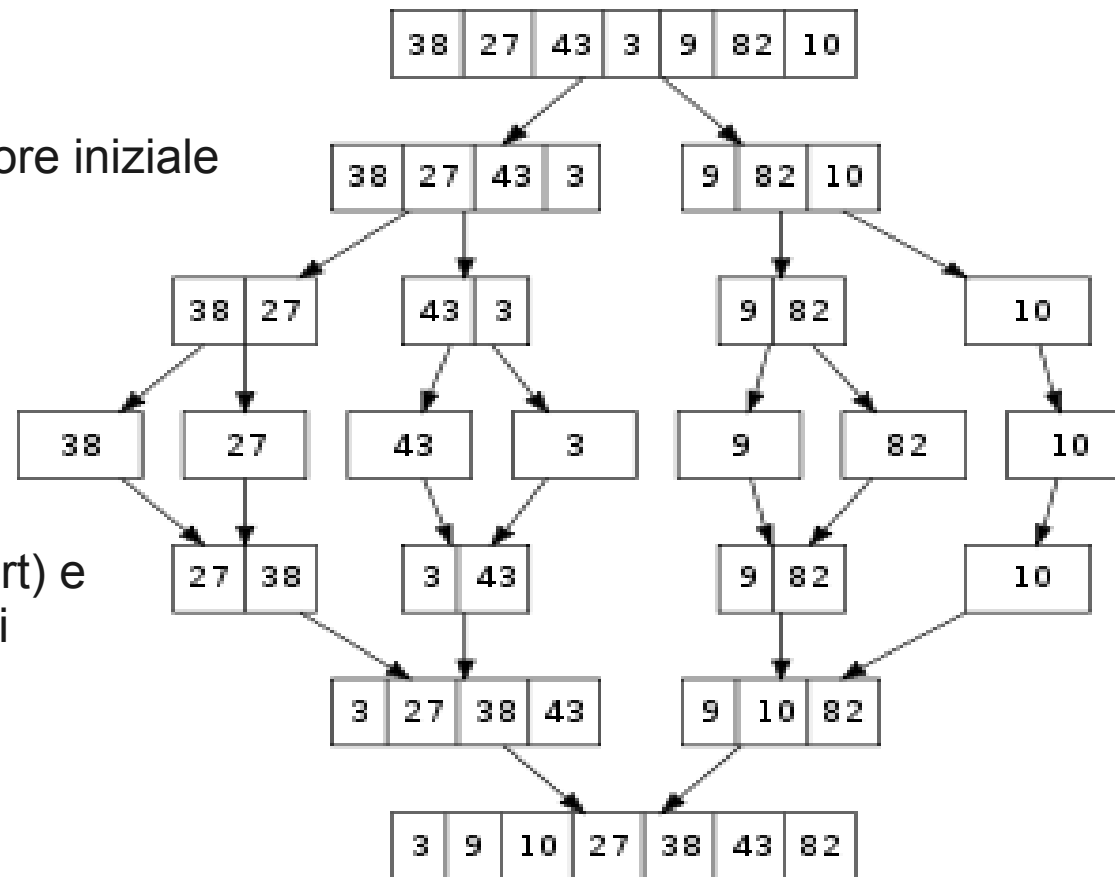
Lezione 6

Merge sort

Divido a metà il vettore iniziale

Metà della metà...

Merge... riordino (sort) e
riassemblo (merge) i
pezzi...



Lezione 6

Merge sort - prestazioni

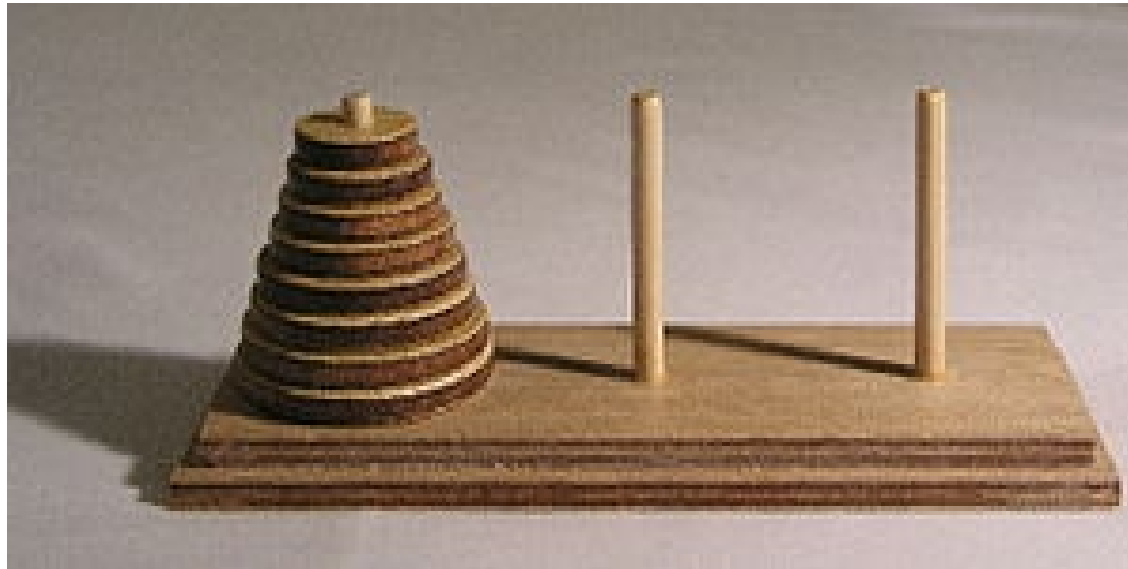
- Tempo di esecuzione del merge dell'ordine di n
- mergesort richiama se stessa due volte ogni volta su metà della porzione di input



$$T(n) = 2T\left(\frac{n}{2}\right) + \Theta(n)$$

Lezione 6

Torre di Hanoi



Lezione 6

Torre di Hanoi

Si scriva un programma per risolvere il rompicapo della Torre di Hanoi: un numero n di dischi, di diametro via via decrescente, sono impilati sul primo dei tre pioli della Torre di Hanoi. L'obiettivo è spostare tutti i dischi sull'ultimo piolo, rispettando le seguenti regole:

- si può spostare un disco solo per volta, su un qualsiasi piolo;
- non si può impilare un disco su uno più piccolo.

Suggerimento: per spostare k dischi dal piolo 1 al piolo 2, bisogna prima spostare i $k-1$ dischi superiori dal piolo 1 al piolo 3.

Questo esercizio può essere risolto in modo “semplice” utilizzando la ricorsione.

Lezione 6

Homework

- Si scriva una versione ricorsiva della successione di Fibonacci:

$$F_0 = 0$$

$$F_1 = 1$$

$$F_n = F_{n-1} + F_{n-2}$$

- Si riscriva l'implementazione dell'algoritmo per il calcolo del MCD in modo ricorsivo;
- Si riscriva la funzione che implementa l'algoritmo *binary search* in forma ricorsiva.