

Lezione 6 - Note

Cosa fa un ordinamento per inserzione?

Chopin, Verdi, Vivaldi | Mozart, Bach, Brahms, Wagner, Debussy, Liszt

Chopin, Mozart, Verdi, Vivaldi | Bach, Brahms, Wagner, Debussy, Liszt

Bach, Chopin, Mozart, Verdi, Vivaldi | Brahms, Wagner, Debussy, Liszt

Bach, Brahms, Chopin, Mozart, Verdi, Vivaldi | Wagner, Debussy, Liszt

Avessimo fatto per selezione avremmo preso Bach, Brahms e Debussy.

Conversione da base 2 a base 16?

$(1011)(0010) = (8+2+1)(2) = B2_{16}$

Non è 112_{16} , che vuol dire $1 \times 256 + 1 \times 16 + 2$, non $11 \times 16 + 2$.

Complessità degli algoritmi?

Se dico $O(1)$, o costante, vuol dire che il tempo non dipende dalla dimensione dell'input.

Se dico $O(n)$, o lineare, vuol dire che il tempo cresce proporzionalmente con la dimensione dell'input.

Se dico $O(n \log n)$, vuol dire che il tempo cresce proporzionalmente a $n \log n$ se la dimensione dell'input è n .

Se dico $O(n^2)$, o quadratico, vuol dire che il tempo cresce con il quadrato della dimensione dell'input.

In che modo capire le complessità degli algoritmi che conosciamo?

Per gli ordinamenti bubble, selezione, inserzione la complessità è quadratica perché servono n passate, ciascuna che mette a posto un elemento, e ogni passata richiede n operazioni fondamentali.

Invece per quick e merge sort? Ogni volta spezzo a metà, quindi devo spezzare $\log n$ volte:

```

                32
           16           16
        8       8       8       8
     4   4   4   4   4   4   4   4
  2  2  2  2  2  2  2  2  2  2  2  2  2  2  2  2
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
```

Codifica delle istruzioni?

Decodificare AC3E12 (sequenza di byte, e quindi in base 16): innanzitutto $AC_{16} \Rightarrow$ moltiplicazione, poi $3E_{16} = 3 \times 16 + 14 = 62$, poi $12_{16} = 1 \times 16 + 2 = 18$. Quindi 62×18 .

Notazione posizionale: $312_b = 3 \times b^2 + 1 \times b^1 + 2 \times b^0$.

Problema algoritmico?

I problemi algoritmici devono avere una definizione precisa, per cui si può controllare se la risposta è quella giusta. In alcuni problemi (come il problema della fermata), si può trovare algoritmi che rispondono in alcuni casi, ma non si possono trovare algoritmi che rispondono in tutti i casi, allora il problema è irrisolvibile.

Architetture multi-core?

Solo da sapere che ogni core esegue istruzioni indipendentemente, in parallelo, e ha i suoi registri e cache. E il fatto che si è passati a mettere più core perché non si poteva rendere più veloci i singoli core.

Qual è il massimo valore che rende vera un'espressione?

Massimo valore di $(zxwxy)_2$ che rende vera $((x \text{ AND } y) \text{ XOR } (z \text{ OR } w)) \text{ AND } (x \text{ XOR } (\text{NOT } w))$.

Chiamiamo $e = ((x \text{ AND } y) \text{ XOR } (z \text{ OR } w))$.

z	x	w	y	xAy	zOw	e	Nw	xXNw	eAxXNw	
0	0	0	0	0	0	0	1	1	0	
0	0	0	1	0	0	0	1	1	0	
0	0	1	0	0	1	1	0	0	0	
0	0	1	1	0	1	1	0	0	0	
0	1	0	0	0	0	0	1	0	0	
0	1	0	1	1	0	1	1	0	0	
0	1	1	0	0	1	1	0	1	1	⇒ minimo, prima riga
0	1	1	1	1	1	0	0	1	0	
1	0	0	0	0	1	1	1	1	1	
1	0	0	1	0	1	1	1	1	1	
1	0	1	0	0	1	1	0	0	0	
1	0	1	1	0	1	1	0	0	0	
1	1	0	0	0	1	1	1	0	0	
1	1	0	1	1	1	0	1	0	0	
1	1	1	0	0	1	1	0	1	1	⇒ massimo, ultima riga
1	1	1	1	1	1	0	0	1	0	

Il minimo è $01110_2 = 8 + 4 + 2 = 14$.

Il massimo è $11110_2 = 16 + 8 + 4 + 2 = 30$.