Analisi – Ordinamento funzioni

Ordinare le seguenti funzioni in accordo alla loro complessità asintotica. Si scriva $f(n) < g(n)$ se $O(f(n)) \subset O(g(n))$. Si scriva $f(n) = g(n)$ se $O(f(n)) = O(g(n))$, ovvero se $f(n) = \Theta(g(n))$.

$$
\begin{aligned}
f_1(n) &= 2^{n+2} \\
f_2(n) &= \log^2 n \\
f_3(n) &= \log_n(n \cdot (\sqrt{n})^2) + \frac{1}{n^2} \\
f_4(n) &= 3n^{0.5} \\
f_5(n) &= 16^{n/4} \\
f_6(n) &= 2\sqrt{n} + 4n^{1/4} + 8n^{1/8} + 16n^{1/16} \\
f_7(n) &= \sqrt{(\log n)(\log n)} \\
f_8(n) &= \frac{n^3}{(n+1)(n+3)} \\
f_9(n) &= 2^n
\end{aligned}
$$

Analisi – MergeSortK

Si consideri una variante di MergeSort chiamata MergeSortK che, invece di suddividere l'array da ordinare in 2 parti, lo suddivide in k parti, ri-ordina ognuna di esse applicando ricorsivamente MergeSortK, e le riunifica usando un'opportuna variante MergeK di Merge, che fonde k sottoarray invece di 2.

(1) Abbozzate il codice di MergeSortK e di MergeK (fatevi solo un'idea, ci sono molti dettagli nella gestione degli indici)

(2) Scrivete la relazione di ricorrenza di MergeSortK

Ripasso

Ripasso del metodo dell'albero di ricorsione

Albero di ricorsione su MergeSortK

Calcolate la complessità computazionale delle seguenti funzioni, utilizzando il metodo dell'albero di ricorsione

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

$$T(n) = \begin{cases} 3T(n/3) + 3n & n > 1 \\ 1 & n = 1 \end{cases}$$

$$T(n) = \begin{cases} kT(n/k) + kn & n > 1 \\ 1 & n = 1 \end{cases}$$

Analisi – Algoritmo di selezione deterministico

Si consideri la seguente equazione di ricorrenza:

$$T(n) = \begin{cases} T(\lfloor n/5 \rfloor) + T(\lfloor 7n/10 \rfloor) + \frac{11}{5}n & n > 1 \\ 1 & n \leq 1 \end{cases}$$

Individuare limiti inferiori e superiori tramite il metodo di sostituzione.

Ricorrenza $2T(n/8) + 2T(n/4) + n$

Trovare un limite asintotico superiore e un limite asintotico inferiore alla seguente ricorrenza, facendo uso del metodo di sostituzione:

$$T(n) = \begin{cases} 1 & n \leq 1 \\ 2T(n/8) + 2T(n/4) + n & n > 1 \end{cases}$$