Università degli Studi di Udine

Relazione di laboratorio del corso di Algoritmi e Strutture Dati

A.A. 2018/2019

Prof. Alberto Policriti

Enrico Basso
Matr. 127935
basso.enrico.3@spes.uniud.it

1 Definizione del problema

Il problema prevede come input una serie di n valori razionali positivi dei quali è necessario individuare la mediana inferiore pesata. Si definisce mediana inferiore pesata di $w_1, ..., w_n$, il peso w_k tale che:

$$\sum_{w_i < w_k} w_i < W/2 \leq \sum_{w_i \leq w_k}$$

2 Soluzione del problema

Per individuare la mediana inferiore è stata ideata una soluzione che si basa sui principi della ricerca binaria e della selezione, entrambi algoritmi *divide et impera*. Il processo di ricerca della mediana inferiore consiste nel verificare se l'elemento che sarebbe posizionato a metà della serie di numeri razionali positivi se essa fosse ordinata soddisfi la relazione descritta nel problema. Per fare questo dobbiamo prima di tutto utilizzare la selezione per cercare l'elemento che sarebbe posizionato a metà della serie, in questo modo i valori w_i vengono divisi in due regioni: $w_i < w_k$ e $w_i \geq w_k$ con w_k uguale all'elemento cercato tramite la selezione. È poi necessario calcolare le sommatorie degli elementi $\sum_{w_i < w_k} w_i$ e $\sum_{w_i \leq w_k} w_i$ per verificare se ffettivamente w_k sia soluzione del problema. In caso negativo è necessario effettuare lo stesso controllo a destra o a sinistra di w_k in base alle sommatorie appena calcolate: se esse sono maggiori di $W/2$ allora bisognerà controllare a sinistra, in caso contrario a destra. Bisogna ripetere il procedimento fino a che non si individui l'elemento mediana inferiore.

L'utilizzo dell'algoritmo di selezione permette la ricerca della mediana inferiore senza dover ordinare del tutto la serie ogni volta, questo comporta un leggero miglioramento dei tempi di esecuzione nel caso in cui la mediana si trovi """"all'inizio"""". Questa soluzione funziona in $O(n \log n)$ perché il costo dell'utilizzo dell'algoritmo di selezione è di $\Theta(n)$ ...

La presente soluzione ha la seguente equazione di complessità:

$$T(n) = \begin{cases} \Theta(1), & n = 1 \\ T(\frac{n}{2}) + O(n), & n > 1 \end{cases}$$

$$= \sum_{i=0}^{\log_2} n - 1dn + cn = dn \log_2 n + cn = O(n \log_2 n)$$

2.1 Implementazione della soluzione

Spiegare select che usa partition con il mediano dei mediani (vedi quaderno). Spiegare anche il fattore random (non ordino sempre un algoritmo, al contrario potrei avere culo e fare meno passi di ricorsione rispetto a un ordinamento fisso).

3 Eseguire il codice

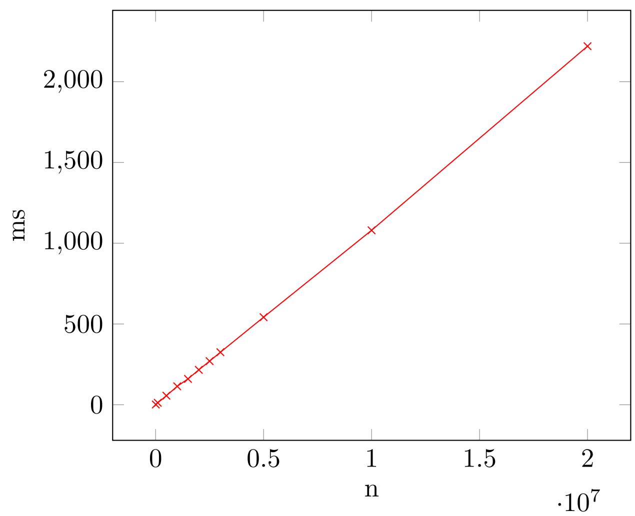
3.1 Caratteristiche della macchina

3.2 Calcolo dei tempi

Per il calcolo dei tempi sono stati utilizzati gli algoritmi visti a lezione, senza però calcolare il tempo per la lettura dell'input: i tempi calcolati sono di sola risoluzione del problema, ovvero con il testo già letto da standard input. I parametri sono stati settati con i seguenti valori:

- K = 5
- $z(1 - \frac{\infty}{2}) = 2.33$
- $\Delta = \frac{1}{5}$ del tempo medio

3.2.1 Caso pessimo e caso medio



3.2.2 Caso ottimo

caso ottimo quando select trova subito il mediano (array con elementi tutti uguali)