

Fondamenti di Informatica, A.A. 2013-2014 - Compito A

22/09/2014

Prova Pratica

Il metodo di ordinamento *quicksort* su vettore (numerico) può essere specificato come segue:

- Si prenda un elemento del vettore come riferimento;
- Si separino tutti gli elementi del vettore di ingresso minori di quello di riferimento da quelli maggiori o uguali;
- Il vettore ordinato è dato dal vettore degli elementi minori, ordinato con una chiamata ricorsiva, seguito dall'elemento di riferimento, seguito dal vettore degli elementi maggiori o uguali escluso quello di riferimento, ordinato con una chiamata ricorsiva.

Si scriva una funzione Matlab/Octave che realizzi questo algoritmo per vettori numerici. Quale è la base della ricorsione?

Svolgimento

La base della ricorsione è data dal fatto che un vettore che contenga 0 o 1 elemento è già ordinato. Va prestata attenzione alla eliminazione dai vettori ausiliari dell'elemento di riferimento (pivot); altrimenti si possono avere sia delle duplicazioni che dei problemi di terminazione. Infatti, la proprietà di terminazione si basa sul fatto che la dimensione dei vettori ausiliari diminuisca a ciascun passo di almeno uno; se come elemento di riferimento si sceglie il minimo, e non lo si elimina dal vettore degli elementi maggiori o uguali, questo vettore sarà della stessa dimensione del vettore di ingresso, e quindi non si progredirà ulteriormente.

La versione presentata qui è organizzata con una funzione *wrapper* che gestisce le condizioni di errore e garantisce che la funzione ricorsiva interna debba gestire solo il caso di vettori riga. Nella funzione ricorsiva poi, si estraggono esplicitamente tutti gli elementi che sono uguali al pivot; questo garantisce la terminazione, anche se comporta una certa ridondanza; il codice sfrutta le operazioni su vettori di Matlab/Octave. Una implementazione efficiente dell'algoritmo dovrebbe usare una strategia di scelta del pivot più sofisticata, in modo da migliorare il bilanciamento dei vettori nelle chiamate ricorsive.

```

function vs=quicksort(v)
% Quicksort: sort a numeric vector by the quicksort method

    if (~isnumeric(v))
        error('Sort_numeric_values_only!');
    end
    [nr,nc]=size(v);
    if ((nr>1)&&(nc>1))
        error('Sort_a_vector_only!');
    end
    if (nr>1)
        v=v';
    end

    vs=InnerQuickSrt(v);
    if (nr>1)
        vs = vs';
    end
end

function vs=InnerQuickSrt(v)

    n=length(v);
    if (n<=1)
        vs = v;
        return
    end

    item = v(1);

    idxl=(v<item);
    idxe=(v==item);
    idxu=(~idxl)&(~idxe);

    vs=[InnerQuickSrt(v(idxl)),v(idxe),InnerQuickSrt(v(idxu))];
end

```