

Fondamenti di Informatica, A.A. 2012-2013

23/09/2013

Esercizio 1

È dato il frammento di codice Matlab

```
v = [1 2 3 4];  
  
for i=1:length(v)  
    v = [v sum(v)];  
    l = length(v);  
    k = l-i;  
    v(k) = v(end);  
    v = v(1:end-1);  
end  
  
disp(v)
```

Si chiede cosa viene visualizzato dall'interprete Matlab.

Soluzione

L'interprete Matlab stampa

```
56      29      16      10
```

Esercizio 2

Discutere come vengono rappresentate le stringhe in MATLAB/Octave, come è possibile manipolarle (concatenazione, troncamento, ecc...) e compararle. Avendo la definizione `str = '2013'` come è possibile estrapolare il valore 2013 in modo da trattarlo come una variabile numerica? Riportare un esempio.

Soluzione

Una stringa è rappresentata come un array di caratteri. Questa definizione consente di manipolare le stringhe come dei comuni arrays; è infatti possibile concatenare le stringhe `a` e `b` riportando il risultato nella variabile `c` con il seguente comando

```
c = [a b]
```

Essendo le stringhe degli array è errato compararle direttamente come si farebbe con variabili numeriche, in quanto il confronto tra array richiede che essi siano delle stesse dimensioni. Per poter effettuare il confronto lessicografico di due stringhe è necessario utilizzare apposite funzioni di MATLAB/Octave tipo `strcmp()`. Un numero scritto all'interno di una stringa non può essere trattato direttamente da MATLAB come una variabile numerica ma ha bisogno di una conversione. Tale conversione può essere effettuata attraverso l'utilizzo della funzione `sscanf()`. Un possibile esempio di utilizzo è il seguente:

```
anno = sscanf(str, '%d')
```

Un'altra funzione presente in MATLAB molto simile alla `sscanf()` è la `str2num()`.

Esercizio 3

In una prova di esame di informatica viene richiesto di realizzare un programma ricorsivo per verificare se una stringa è palindroma o meno. Una stringa si dice palindroma se è uguale sia letta da sinistra verso destra che da destra verso sinistra (es: 'rotor'). Uno studente propone la seguente soluzione:

```
function res = palindrome(str)

    len = length(str);

    res = 1;

    if(len==1)
        return;
    end

    res = res * palindrome(str(2:len-1));

    if(str(1) ~= str(len))
        res = 0;
        return
    end

end
```

Si chiede di trovare l'errore presente nel codice e di proporre una correzione adeguata. La funzione in questione ritorna 1 se la stringa passata è palindroma o zero altrimenti.

Soluzione

L'errore è relativo alla verifica del raggiungimento del passo base; infatti, ad ogni chiamata ricorsiva la lunghezza della stringa viene diminuita di 2, e quindi se una stringa è composta da un numero pari di caratteri tale funzione produce un errore poichè non capita mai che la lunghezza della stringa sia uguale ad 1. Una possibile soluzione è la seguente:

```
function res = palindrome(str)

    len = length(str);

    res = 1;

    if(len==1 || len==0)
        return;
    end

    res = res * palindrome(str(2:len-1));
```

```

    if (str(1) ~= str(len))
        res = 0;
        return
    end
end

```

end

Esercizio 4

È data la seguente funzione Matlab

```

function [res]=mystery(m,x,y,k)
    for i=1:(g(k)/2)
        y = y + m^x;
    end
    res=y
end

```

Si stimi il numero di operazioni aritmetiche eseguito dal codice, considerando le seguenti possibili scelte di $g(p)$

1. $g(p) = 4$;
2. $g(p) = 2 * p$;
3. $g(p) = 6 * p$.

Si assuma che m sia una matrice quadrata $n \times n$, che x sia un intero positivo e y sia uno scalare.

Soluzione

L'elevamento a potenza intera x di una matrice può essere realizzato con $x - 1$ moltiplicazioni matrice-matrice. Ciascuna di queste operazioni, per una matrice $n \times n$, costa $2 * n^3$ operazioni aritmetiche. Poichè viene sommato uno scalare ad ogni elemento della matrice risultante si hanno ulteriori n^2 operazioni. In totale, per una singola iterazione del ciclo, si avranno $2(x - 1)n^3 + n^2$ operazioni aritmetiche. Nel caso in cui $g(i) = 4$ si avranno in totale $2 * (2(x - 1)n^3 + n^2)$ operazioni. Nel caso in cui $g(i) = 2 * i$ si avranno $k * (2(x - 1)n^3 + n^2)$ operazioni. Nel caso in cui $g(i) = 6 * i$ si avranno $3 * k * (2(x - 1)n^3 + n^2)$ operazioni.

È da notare che la realizzazione della potenza con una sequenza di $x - 1$ moltiplicazioni non è necessariamente la strategia più economica; ad esempio se $x = 2^k$ si può usare una strategia di elevazioni al quadrato successive per un totale di $k = \log_2(x)$ moltiplicazioni matrice-matrice.

Esercizio 5

È dato il frammento di codice Matlab

```

ii = 1; jj = 0; kk = 0; hh = 0;

while(ii<=length(a))
    switch mod(a(ii),2)
        case 0
            jj = 12+ii;
        otherwise
            switch mod(a(ii),3)
                case 1
                    jj = 11+ii;
                otherwise
                    kk = 11+ii;
                    switch mod(jj,12)
                        case 0
                            hh = hh+1;
                    end
                end
            end
        end
    end
    ii = ii+1;
end

```

Riscrivere il codice facendo uso di uno o più cicli `for` ed sostituendo i costrutti `switch` con altri appropriati.

Soluzione

Una possibile soluzione è la seguente:

```

ii = 1; jj = 0; kk = 0; hh = 0;
for ii = 1:length(a)
    if mod(a(ii),2)==0
        jj = 12+ii;
    elseif mod(a(ii),3)==1
        jj = 11+ii;
    else
        kk = 11+ii;
        if mod(jj,12)==0
            hh = hh+1;
        end
    end
end
end

```