Esame di Programmazione Informatica

24 febbraio 2021

Informazioni generali

- 2 esercizi di script MATLAB e 2 domande a risposta multipla.
- La durata dell'esame è di 2 ore.
- È consigliato l'uso di MATLAB nello svolgimento di tutti i punti.
- Le slide e gli script del corso sono ovviamente consultabili liberamente.
- Il punteggio assegnato ad ogni esercizio ed alle domande a risposta multipla è indicato tra parentesi.

Esercizio 1 (14/30)

Dati due polinomi $a(x)$ e $b(x)$ di secondo grado

$$a(x) = a_0 + a_1 x + a_2 x^2,$$
$$b(x) = b_0 + b_1 x + b_2 x^2,$$

essi possono essere rappresentati dai vettori riga $\mathbf{a}$ e $\mathbf{b}$ dei loro coefficienti:

$$\mathbf{a} = [a_0, a_1, a_2],$$
$$\mathbf{b} = [b_0, b_1, b_2].$$

Il prodotto dei due polinomi è il polinomio di quarto grado $p(x)$

$$\begin{aligned} p(x) = a(x)b(x) &= \left(a_0 + a_1 x + a_2 x^2\right)\left(b_0 + b_1 x + b_2 x^2\right) = \\ &= (a_0 b_0) + \\ &+ (a_0 b_1 + a_1 b_0)x + \\ &+ (a_0 b_2 + a_1 b_1 + a_2 b_0)x^2 + \\ &+ (a_1 b_2 + a_2 b_1)x^3 + \\ &+ (a_2 b_2)x^4 \end{aligned}$$

che può essere nuovamente rappresentato dal vettore riga $\mathbf{p}$ dei suoi 5 coefficienti, ottenuti dalla precedente formula:

$$\mathbf{p} = [p_0, p_1, p_2, p_3, p_4].$$

Scrivere una funzione MATLAB che prenda in ingresso due vettori riga **a** e **b** dei coefficienti di $a(x)$ e $b(x)$ e restituisca in uscita il vettore riga **p** dei coefficienti del prodotto $p(x) = a(x)b(x)$.

Si mostri poi come utilizzare la funzione scritta precedentemente per calcolare i coefficienti del prodotto tra $a(x) = 1 + x^2$ e $b(x) = 1 - x^2$.

Soluzione: è sufficiente scrivere esplicitamente la formula che fornisce i coefficienti di $p(x)$, ricordandosi che in MATLAB gli indici dei vettori partono sempre da 1:

prodotto_polinomi_quadratici.m

```
function p = prodotto_polinomi_quadratici( a , b )
   p    = zeros( 1 , 5 ) ;
   p(1) = a(1)*b(1) ;                            % a_0 b_0
   p(2) = a(1)*b(2) + a(2)*b(1) ;                % a_0 b_1 + a_1 b_0
   p(3) = a(1)*b(3) + a(2)*b(2) + a(3)*b(1) ;    % a_0 b_2 + a_1 b_1 + a_2 b_0
   p(4) =             a(2)*b(3) + a(3)*b(2) ;    % a_1 b_2 + a_2 b_1
   p(5) =                         a(3)*b(3) ;    % a_2 b_2
end
```

Utilizzo della funzione scritta precedentemente per calcolare il prodotto di $a(x) = 1+x^2$, cioè $\mathbf{a} = [1, 0, 1]$, e $b(x) = 1 - x^2$, cioè $\mathbf{b} = [1, 0, -1]$:

```
a = [1,0, 1] ; % a(x) = 1 + x^2
b = [1,0,-1] ; % b(x) = 1 - x^2
p = prodotto_polinomi_quadratici( a , b )
p =
      1      0      0      0     -1
```

ottenendo correttamente $\mathbf{p} = [1, 0, 0, 0, -1]$, cioè $p(x) = 1 - x^4$.

Esercizio 2 (14/30)

Date due funzioni $f(x)$ e $g(x)$, la derivata del loro quoziente $q(x) = \dfrac{f(x)}{g(x)}$ è

$$q'(x) = \frac{f'(x)g(x) - f(x)g'(x)}{[g(x)]^2}.$$

Scrivere una funzione MATLAB che prenda in ingresso:

- il *function handle* `f` della funzione $f(x)$,
- il *function handle* `df` della sua derivata $f'(x)$,
- il *function handle* `g` della funzione $g(x)$,
- il *function handle* `dg` della sua derivata $g'(x)$,
- due valori scalari `a` e `b`

e faccia il plot grafico di $q'(x)$ nell'intervallo $[\mathtt{a}, \mathtt{b}]$ suddiviso uniformemente in 1000 punti. La funzione richiesta non ha bisogno quindi di nessun argomento in uscita.

Si mostri poi come utilizzare la precedente funzione per fare il plot della derivata del quoziente delle due funzioni $f(x) = \sin(x)$, la cui derivata è $f'(x) = \cos(x)$, e $g(x) = g'(x) = e^x$ nell'intervallo $[-\pi, \pi]$.

Soluzione: costruiamo il vettore `x` che copre l'intervallo $[\mathtt{a}, \mathtt{b}]$ in 1000 punti equidistanti e calcoliamo il vettore `dq` $= q'(\mathtt{x})$ su tali valori di x utilizzando operazioni elemento per elemento (`.*` , `./` , `.^`). Plottiamo infine il grafico con `plot`:

`plot_derivata_quoziente.m`

```
function plot_derivata_quoziente(f, df, g, dg, a, b)
  x  = linspace(a, b, 1000) ;
  dq = ( df(x).*g(x) - f(x).*dg(x) ) ./ g(x).^2 ;
  plot(x, dq) ;
end
```

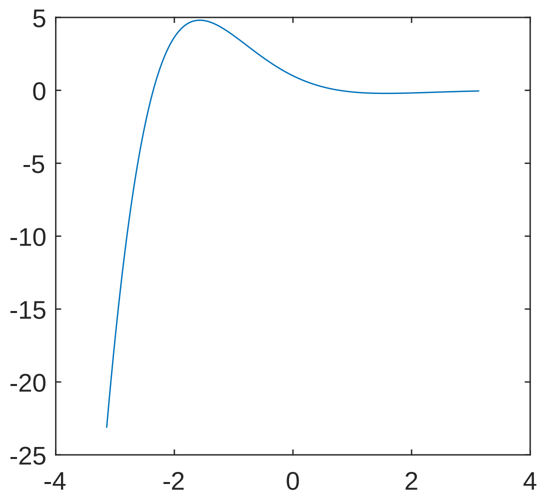
Utilizzo della precedente funzione nel caso di $f(x) = \sin(x)$, $f'(x) = \cos(x)$, $g(x) = g'(x) = e^x$ nell'intervallo $[-\pi, \pi]$:

```
% Definizione di f(x), f'(x) e g(x), g'(x) mediante funzioni anonime
f   = @(x) sin(x) ;
df  = @(x) cos(x) ;
g   = @(x) exp(x) ;
dg  = @(x) exp(x) ;

% Utilizzo nell'intervallo [-π,π]
a  = -pi ;
b  =  pi ;
plot_derivata_quoziente(f, df, g, dg, a, b) ;
```

ottenendo correttamente il seguente grafico della derivata del quoziente $\frac{\sin(x)}{e^x}$, che analiticamente vale:

$$\frac{d}{dx}\frac{\sin(x)}{e^x} = \frac{\cos(x) - \sin(x)}{e^x}$$



Domande a risposta multipla (5/30)

Domanda 1 (2/30)

Quanto spazio è necessario per rappresentare in un calcolatore, senza alcuna compressione, un'immagine raster di dimensione $m \times n$ utilizzando una quantizzazione con 2^k livelli per ciascun canale?

- 500×500 con $2^8 = 256$ livelli, 3 canali RGB: 6 Mbit = 750 kB.
- 1000×1500 con $2^6 = 64$ livelli di grigio: 9 Mbit = 1.125 MB.
- 1000×1000 con $2^4 = 16$ livelli, 3 canali RGB: 12 Mbit = 1.5 MB.

Domanda 2 (3/30)

Caso A. Si consideri la seguente funzione ricorsiva MATLAB:

```
function y = f(x)
  if x > 1
    y = x * f(x/2) ;
  else
    y = x ;
  end
end
```

Richiamando `f(k)` con `k` > 1, come si comporta la precedente funzione?

- Non restituisce nessun valore perchè `f` è ricorsiva.
- Va avanti all'infinito a richiamare se stessa con argomenti `x` sempre più piccoli.
- Restituisce sempre un numero perchè il numero di ricorsioni è sempre limitato.
- Restituisce sempre 1.

Soluzione: restituisce sempre un numero perchè il numero di ricorsioni è sempre limitato. L'argomento `x` della funzione viene diviso per due ad ogni ricorsione, quindi prima o poi si avrà sempre `x` ≤ 1 e la ricorsione si conclude.

Caso B. Si consideri la seguente funzione ricorsiva MATLAB:

```
function y = f(x)
  if x > -1
    y = x * f(x/2) ;
  else
    y = x ;
  end
end
```

Richiamando `f(k)` con `k` > -1, come si comporta la precedente funzione?

- Non restituisce nessun valore perchè `f` è ricorsiva.
- Va avanti all'infinito a richiamare se stessa con argomenti `x` sempre più piccoli.
- Restituisce sempre un numero perchè il numero di ricorsioni è sempre limitato.

- Restituisce sempre `-1`.

Soluzione: va avanti all'infinito a richiamare se stessa con argomenti `x` sempre più piccoli. L'argomento `x` della funzione viene diviso per due ad ogni ricorsione avvicinandosi sempre di più a 0 ma la ricorsione non si arresterà mai in quanto si verificherà sempre $\texttt{x} > -1$.