ESERCITAZIONE MATLAB 2: M-files

1. Scrivere un M-file di tipo FUNCTION che prenda in ingresso una generica matrice `A` e restituisca in uscita una matrice `B` delle stesse dimensioni di `A` i cui elementi siano il risultato della valutazione della seguente funzione su ciascuna componente di A:

   $$f(x) = \sin(x)^2 \cos(x).$$

   Salvare il file con il nome `funzionef.m.` Se ne verifichi il funzionamento eseguendo da prompt dei comandi

   ```
   funzionef(linspace(0,pi,5))
   ```

   Si verifichi quindi che si ottiene lo stesso risultato eseguendo

   ```
   feval('funzionef',linspace(0,pi,5))
   ```

2. Si scriva un M-file di tipo function per il calcolo delle radici reali del polinomio di secondo grado

   $$p(x) = x^2 + 2\,b\,x + c.$$

   Il codice deve prendere in ingresso i due coefficienti `b` e `c` e fornire in uscita un vettore contenente le due radici. Nel caso in cui il polinomio non abbia radici reali il calcolo deve essere interrotto con un messaggio di errore.

3. Scrivere un M-file di tipo FUNCTION che prenda in ingresso due parametri `m` e `n` e restituisca in uscita una matrice `A` di dimensione $m \times n$ i cui elementi siano dati da

   $$a_{ij} = \begin{cases} 0 & \text{se } i = j, \\ 1/(i-j) & \text{se } i > j, \\ 1/(j-i) & \text{se } j > i. \end{cases}$$

4. La successione di Fibonacci $\{f_n\}_{n=0,1,2,\dots}$ è una successione divergente così definita

   $$\begin{cases} f_0 = 1, \\ f_1 = 1, \\ f_n = f_{n-1} + f_{n-2}, \qquad n \geq 2. \end{cases}$$

   Si scriva un M-file di tipo FUNCTION che prenda in ingresso un valore $x \geq 1$ e determini il più piccolo n tale che $f_n \geq x$.

5. Si scriva un M-file di tipo SCRIPT contenente le seguenti istruzioni.

   ```
   x=0;
   while (x~=1)
      x = x + 0.1,
   end
   ```

   Si provi ad eseguirlo e si spieghi il suo non funzionamento.

6. Scrivere un M-file di tipo SCRIPT per disegnare le due funzioni:

   $$y_1 = x^2 \cos x \qquad y_2 = x^2 \sin x$$

   sullo stesso grafico. Prendere $x = -2{:}0.1{:}2$. Assegnare all'asse delle ascisse l'etichetta `x` e al disegno il titolo `grafico di y1 e y2`.

SOLUZIONI:

1. ```
   function B = funzionef(A)
    B = sin(A).^2.*cos(A);
   ```

2. ```
    function x = radici(b,c)
   %   x = radici(b,c)
   %
   %   calcola le radici reali del polinomio di secondo grado
   %
   %   x^2 + 2*b*x + c = 0
   %
   %   Input: b,c coefficienti del polinomio
   %   Output: vettore x contenente le due radici del polinomio

   if (c==0)
       x = [0 -2*b];
   else
       delta = b^2-c;
       if (delta<0)
           error('il polinomio non ammette radici reali')
       end
       if (b>=0)
           x(1) = -b-sqrt(delta);
       else
           x(1) = -b+sqrt(delta);
       end
       x(2) = c/x(1);
   end
   ```

3. ```
   function A = matrice(m,n)

   A = zeros(m,n);
   for i=1:m
     for j=1:n
         if (i>j)
             A(i,j) = 1/(i-j);
         elseif(j>i)
             A(i,j) = 1/(j-i);
         end
     end
   end
   ```

4. ```
    function n = fibonacci(x)

   % n = fibonacci(x)
   %
   % Determina il piu' piccolo n tale che fn >=x dove fn e'
   % l'ennesimo termine della successione di Fibonacci.

   fnm1 = 1;
   if (fnm1>=x),
   ```

```
    n=0;
    return,
end

n  = 1;
fn = 1;

while (fn<x)
    fnp1 = fn+fnm1;
    fnm1 = fn;
    fn   = fnp1;
    n    = n+1;
end

if (fn==Inf),
    error('Overflow: impossibile determinare n')
end
```

5. Se le operazioni aritmetiche fossero svolte in aritmetica esatta, dopo 10 iterazioni del ciclo `while` si avrebbe `x` = `1` e questo porterebbe alla chiusura del ciclo. Questo non accade, invece, in aritmetica finita (si ricordi anche che 0.1 non è un numero macchina per lo standard IEEE 754 per la doppia precisione). Pertanto, il valore della variabile `x` risulta essere diverso da 1 ad ogni iterazione e di conseguenza la esecuzione del ciclo `while` non viene mai arrestata.

6.
```
% SCRIPT file disegno.m
x = -2:0.1:2;
y1 = x.^2.*cos(x);
y2 = x.^2.*sin(x);
plot(x,y1,'g-*',x,y2,'r-o')
xlabel('x')
title('grafico di y1 e y2')
```