Terza Esercitazione: Cicli in Matlab - Schema di Hörner e bisezione

Esercizio 1 *Tabulazione di un polinomio mediante schema di Ruffini-Hörner*

Uso del ciclo **`for...end`**

Costruire un m-file funzione avente i seguenti argomenti in entrata:

`q`	polinomio sotto forma di vettore	**`a`**	primo estremo dell'intervallo
`b`	secondo estremo dell'intervallo	**`p`**	punti

La funzione tabulerà la funzione polinomiale in **`[a,b]`** suddiviso in **`p`** punti (funzione **`linspace`**) usando lo schema di Ruffini-Hörner.

Quindi in uscita avrà una matrice a due colonne. Nella prima colonna le **`x`**, nella seconda le **`y`**.

Il vettore **`q`** verrà inteso come polinomio nel seguente modo: Se **`q=[q(1),q(2),...,q(n)]`**, il polinomio è

$$q(x) = q(1)\cdot x^{n-1} + q(2)\cdot x^{n-2} + \cdots + q(n-1)\cdot x + q(n)$$

La notazione non usuale dipende dal fatto che l'indice 0 non è ammesso nelle array e che conviene (per compatibilità future) scrivere il polinomio partendo dal coefficiente di grado maggiore.

Per Ruffini-Hörner

$$q(x) = \Big(\Big(\Big(\cdots\Big(q(1)\cdot x\Big) + q(2)\Big)\cdot x + q(3)\Big)\cdot x + \cdots\cdots + q(n-1)\Big)\cdot x + q(n)$$

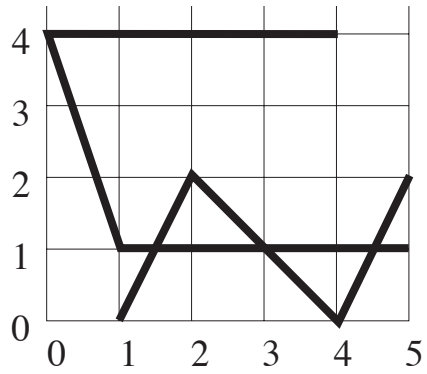
Per procedere per passi successivi, inizialmente semplifichiamo il file e facciamo in modo che calcoli **`q`** solo in **`a`** (e ignoriamo **`b`** e **`n`**). L'output sarà la coppia **`(a,q(a))`**.

```
function t=RH(q,a,b,p)
% tabula il polinomio q calcolato in p punti in [a,b]
x=linspace(a,b,p); % questo servirà in un secondo tempo
...% calcolo del grado di q
y=0 % inizializziamo il calcolo
      for idx=1:...
          y=y...
      end
```

Quando la **`function`** sarà a posto e saremo sicuri che funzioni, la amplieremo in modo che faccia quanto richiesto

Esercizio 2 *Grafica in Matlab* **Spezzate**

Disegnare le due spezzate.
(comando **`plot`**)



Esercizio 3 *Grafici in Matlab* **Grafici di funzioni**

Scegliere un passo p e disegnare il grafico delle due funzioni negli intervalli dati:

$f(x) = \dfrac{x^2+x+1}{x^2+1}$ $\quad x \in [-2,2]$	$f(x) = \log(x^2-x+1)$ $\quad x \in [0,2]$

Scegliere un passo p e disegnare il grafico della funzione

$$f(x) = x^7 - 10x^5 + 3x + 1$$

nell'intervallo $[-1,1]$ usando la **`function`** **`R-H`**

./.

Esercizio 4 *Metodo di bisezione*

Uso di `while...end` o `for...end` e di funzione esterna

Consideriamo la funzione $f(x) = \dfrac{1}{x^2 + 1/2} - e^x$ nell'intervallo $[0, 1]$.

1. Disegnare il grafico di $f(x)$ in $[0, 1]$
2. Verificare, usando il teorema degli zeri, che esiste un punto $\alpha \in [0, 1]$ tale che $f(\alpha) = 0$. Dire perché il punto è unico.
3. Determinare analiticamente il numero N di iterazioni necessarie per calcolare α con un errore inferiore a $\varepsilon = 10^{-10}$ con il metodo di bisezione.
4. Completare il seguente script che implementa il metodo di bisezione, inserendo istruzioni al posto dei puntini

```
a = 0; b = 1; % intervallo iniziale
epsilon = 1e-10; % errore massimo
h = b-a;
x = (a+b)/2;
```

con un ciclo `for...end`

```
N=........
for idx=1:N
......
```

È bene creare un file esterno `'funzione.m'` contenente la funzione su cui lavorare, che venga richiamato dallo script.

```
function y = funzione(x)
y = 1/(x^2+0.5)-exp(x);
```

5. Stesso inizio, ma senza usare N, mediante un ciclo `while...end` e calcolando con un contatore `NumIt` il numero di iterazioni.

```
NumIt = 0;
while (h>epsilon)
....
      if NumIt>100;
            disp('troppe iterazioni');
      end
end
```