III esercitazione di
Laboratorio Computazionale Numerico

1. Dato il vettore x di elementi equidistanti $-5, -4, .., 8, 9$ determinare l'elemento massimo, minimo, di valore assoluto massimo, di valore assoluto minimo, la somma di tutti gli elementi e la somma dei valori assoluti di tutti gli elementi.

2. Calcolare e confrontare, mediante una tabella, i valori, al tendere di x a zero, delle due funzioni:
$$f(x) = \frac{(x+1)^2 - 1}{x}, \qquad f(x) = x + 2.$$
Commentare i risultati.

3. Si approssimi il valore di e^{-9}, utilizzando lo sviluppo in serie di e^x:
$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots \tag{1}$$
Quanti termini sono necessari per stabilizzare il risultato con 6 cifre?
Ripetere il calcolo ricordando che possiamo scrivere: $e^{-x} = 1/e^x$. Costruire, per un confronto, una tabella e un grafico degli errori relativi dei valori ottenuti sommando n termini usando i due diversi algoritmi.

4. In relazione al calcolo numerico della derivata di $f(x) = e^x$, in $x = 1$, si considerino le seguenti approssimazioni:
$$\frac{f(x+h) - f(x)}{h}, \quad \frac{f(x+h) - f(x-h)}{2h}.$$
Si indichi l'errore analitico per entrambe le discretizzazioni considerate e si predisponga con Matlab, al variare del parametro $h = 10^{-\alpha}$, $\alpha = 1, \cdots, 20$ una tabella in `format short e` contenente i valori di h e i corrispondenti errori analitici.
Analizzare i risultati ottenuti.

5. Usando il comando Matlab: `hilb(n)` costruire la matrice di Hilbert di ordine 10.
Costruire poi il vettore b in modo che il sistema lineare: $Hx = b$ abbia soluzione il vettore x con tutte componenti uguali ad 1.
Risolvere il sistema lineare $Hx = b$ usando comandi Matlab.
Costruire poi il vettore $c = b + z$ dove $z^t = [0.001 \ 0 \ \cdots \ 0]$ e risolvere il nuovo sistema $Hy = c$.
Calcolare $e_r = \|x - y\|_2 / \|x\|_2$.

```
% Esercizio 1 - Esercitazione 3
clear all
close all
clc


x=[-5:1:9]
% Elemento massimo del vettore x
max_x=max(x)
% Elemento minimo del vettore x
min_x=min(x)
% Valore assoluto dell'elemento massimo del vettore x
max_abs_x=max(abs(x))
% Valore assoluto dell'elemento minimo del vettore x
min_abs_x=min(abs(x))
% Somma di tutti gli elementi
somma=sum(x)
% Somma di tutti i valori assoluti degli elementi
somma_ass=sum(abs(x))
```

COMMAND WINDOW

```
x =

    -5    -4    -3    -2    -1     0     1     2     3     4     5     6     7     8     9


max_x =

     9


min_x =

    -5


max_abs_x =

     9


min_abs_x =

     0


somma =

    30


somma_ass =

    60
```

```
% Esercizio 2 - Esercitazione 3
clear all
close all
clc
format long
% utilizziamo logspace in modo che la x decresca più velocemente
x = logspace(0,-20,10);
y=(((x+1).^2)-1)./x;
y_2=x+2;


res=[y' y_2']

% per x che tende a zero la prima funzione tende a 0, inoltre con x
% sufficientemente piccola il termine (x+1)^2 viene approssimato dalla
% macchina a 1, rendendo il numeratore uguale a 0.
% la seconda funzione tende a 2.
format short
```

COMMAND WINDOW

```
res =

   3.000000000000000   3.000000000000000
   2.005994842503198   2.005994842503189
   2.000035938133053   2.000035938136638
   2.000000215976404   2.000000215443469
   2.000000140943814   2.000000001291550
   2.000015123252162   2.000000000007743
   1.999630906264934   2.000000000000047
   1.595973870307355   2.000000000000000
                   0   2.000000000000000
                   0   2.000000000000000
```

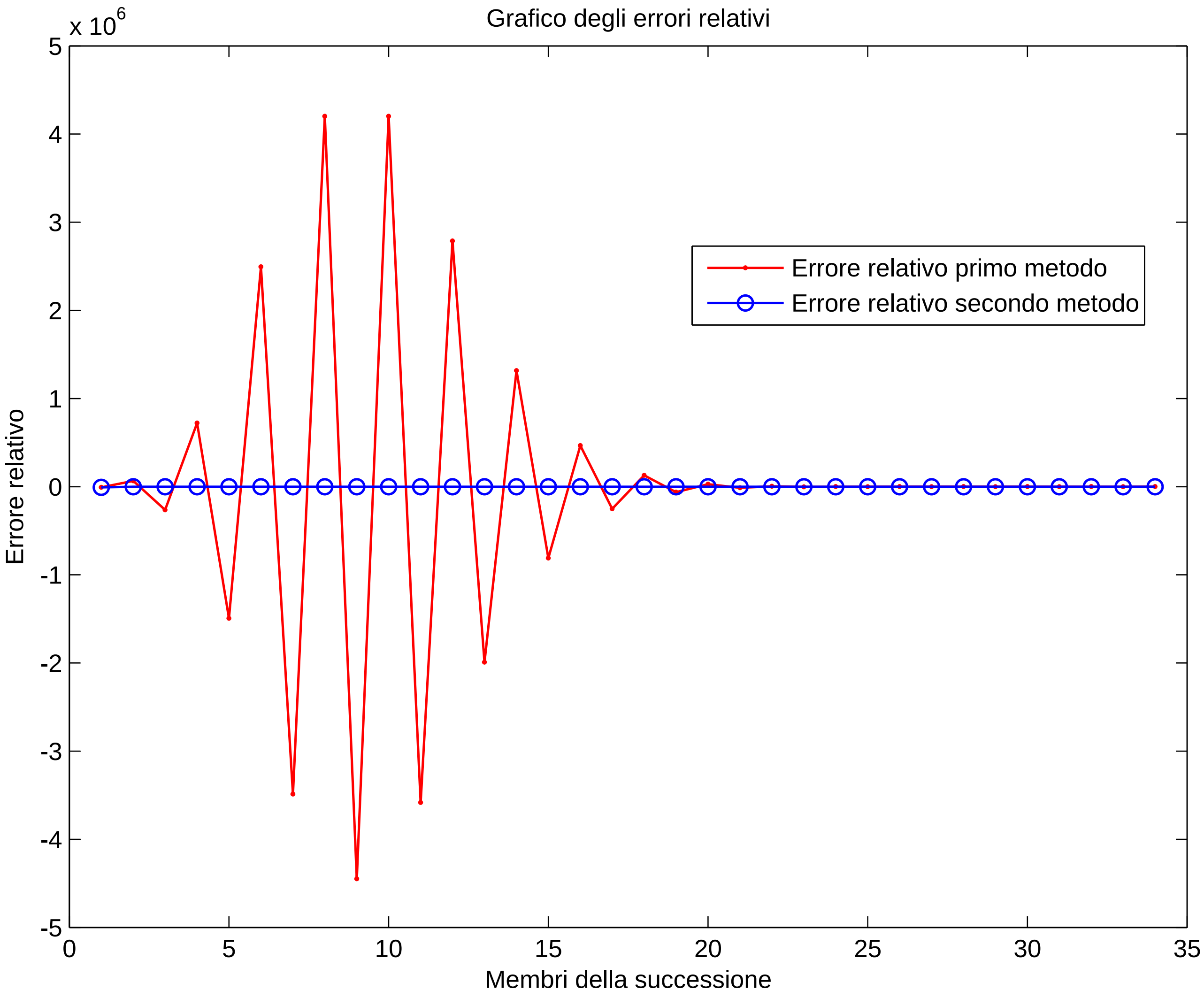
```
% Esercizio 3 - Esercitazione 3
clear all
close all
clc
format long
% Stabilizzare il risultato con 6 cifre significa approssimare il risultato
% con un errore minore o uguale a 10^-6.
% quindi aggiungiamo membri alla serie fino a che l'errore non sia%
inferiore di 10^-6
ve=exp(-9);
x=-9;
serie=1+x;
err= ve-serie;
% tolleranza
t=10^-6;
% i sono i membri della serie utilizzati
i=2;
% aggiungiamo membri fino a che l'errore non sia minore di t
while (abs(err)>t)
    serie = serie + (x^i)/factorial(i);
    err= ve-serie;
    i=i+1;
end
% numero di membri
i
% valore di e^-9 calcolato con la funzione exp di matlab
ve
% valore di e^-9 calcolato con la serie
serie

% ripetiamo il calcolo conderando che e^-x si puo calcolare con 1/e^x

ve_2= 1/exp(9);
x_2=9;
serie_2=1+x_2;
serie_num= 1/serie_2;
err_2= ve_2 - serie_num;

% i sono i membri della serie utilizzati
i_2=2;
% aggiungiamo membri fino a che l'errore non sia minore di t
while (abs(err_2)>t)
    serie_2 = serie_2 + (x_2^i_2)/factorial(i_2);
    serie_num= 1/serie_2;
    err_2= ve_2 - serie_num;
    i_2=i_2 + 1;
end
% numero di membri
i_2
% valore di e^-9 calcolato con la funzione exp di matlab
ve_2
% valore di e^-9 calcolato con la serie
serie_num
% Osservando i metodi utilizzati notiamo che, il primo metodo, cioe e^-x
```

```
% richiede 34 termini invece il secondo metodo, cioè 1/e^x richiede 18
% membri, questo ci permette di dedurre che la convergenza della
% successione con il secondo metodo è piu veloce nello stabilizzare il
% risultato perche richiede un minor numero di membri.

% Costruiamo una tabella per il confronto degli errori relativi dei valori
% ottenuti sommando n termini.

% PRIMO METODO
ve=exp(-9);
x=-9;
serie=1+x;
err= ve-serie;
t=10^-6;
i=2;
% primo elemento primo metodo
result(1,1)=1;
% secondo elemento primo metodo
result(2,1)=serie;
while (abs(err)>t)
    serie = serie + (x^i)/factorial(i);
    err= ve-serie;
    i=i+1;
    result(i,1)=serie;
end
% SECONDO METODO
ve_2=1/exp(9);
x_2=9;
serie_2=1+x_2;
serie_num= 1/serie_2;
err_2= ve_2 - serie_num;
i_2=2;
result(1,2)=1;
result(2,2)=serie_num;
while (i_2<i)
    serie_2 = serie_2 + (x_2^i_2)/factorial(i_2);
    serie_num= 1/serie_2;
    err_2= ve_2 - serie_num;
    i_2=i_2 + 1;
    result(i_2,2)=serie_num;
end
% tabella contenente nella prima colonna i valori del primo metodo e nella
% seconda colonna i valori del secondo metodo
result
% tabella errori relativi
err_rel(:,1)=(ve-result(:,1))./ve;
err_rel(:,2)=(ve_2-result(:,2))./ve_2;
err_rel
% grafico errori relativi
figure();
box on;
n=[1:1:34];
plot(n,err_rel(:,1),'*-',n,err_rel(:,2),'o-')
% dal grafico si può vedere che l'errore relativo del primo metodo oscilla
% mentre con il secondo metodo è più stabile.

format short
```

COMMAND WINDOW Esercizio 3 - Esercitazione 3

```
i = 34

ve =
    1.234098040866796e-04

serie =
    1.226613221354248e-04

i_2 =
    18

ve_2 =
    1.234098040866795e-04

serie_num =
    1.240698022398143e-04

result =

   1.0e+02 *

   0.010000000000000   0.010000000000000
  -0.080000000000000   0.001000000000000
   0.325000000000000   0.000198019801980
  -0.890000000000000   0.000058139534884
   1.843750000000000   0.000022452989054
  -3.077000000000000   0.000010667235586
   4.304125000000000   0.000005968145026
  -5.185892857142858   0.000003810156244
   5.490377232142857   0.000002708418715
  -5.185892857142858   0.000002100920531
   4.422750223214285   0.000001748043141
  -3.438866842532468   0.000001536843285
   2.457345956777597   0.000001409152185
  -1.624647519667833   0.000001332504525
   0.999491143761372   0.000001287485326
  -0.574992054296151   0.000001261904948
   0.310654744611206   0.000001247957747
  -0.158217090104454   0.000001240698022
   0.076218827253376   0.000001237099742
  -0.034829765179280   0.000001235402569
   0.015142101415415   0.000001234640359
  -0.006274412839454   0.000001234313985
   0.002486888446629   0.000001234180519
  -0.000941446839230   0.000001234128300
   0.000344178892967   0.000001234108720
  -0.000118646370624   0.000001234101671
   0.000041562374465   0.000001234099231
  -0.000011840540564   0.000001234098418
   0.000005324682124   0.000001234098156
  -0.000000002455952   0.000001234098075
   0.000001595685471   0.000001234098051
   0.000001131708929   0.000001234098044
   0.000001262202331   0.000001234098042
   0.000001226613221   0.000001234098041

err_rel =

   1.0e+06 *

  -0.008102083927575  -0.008102083927575
   0.064825671420603  -0.000809308392758
  -0.263349227646200  -0.000159457107477
   0.721175469554209  -0.000046110953067
  -1.494005099146711  -0.000017193845473
   2.493319924514945  -0.000007643750523
  -3.487667610977539  -0.000003836038004
   4.202173506084228  -0.000002087401582
  -4.448897750610261  -0.000001194654415
   4.202173506084228  -0.000000702393539
  -3.583790624940812  -0.000000416454028
   2.786543664079675  -0.000000245317013
  -1.991207052685690  -0.000000141847842
   1.316466520459563  -0.000000079739600
  -0.809895062276671  -0.000000043260166
   0.465921887365069  -0.000000022532170
  -0.251725146808410  -0.000000011230636
   0.128205635989315  -0.000000005348020
  -0.061759755409548  -0.000000002432304
   0.028223851042545  -0.000000001057070
  -0.012268771860897  -0.000000000439445
   0.005085209383436  -0.000000000174982
  -0.002014146580155  -0.000000000066832
   0.000763862275163  -0.000000000024520
  -0.000277891045581  -0.000000000008653
   0.000097140149887  -0.000000000002941
  -0.000032678340852  -0.000000000000964
   0.000010594489394  -0.000000000000305
  -0.000003314634614  -0.000000000000093
   0.000001001990078  -0.000000000000028
  -0.000000292997329  -0.000000000000008
   0.000000082966757  -0.000000000000002
  -0.000000022773142  -0.000000000000001
   0.000000006065012  -0.000000000000000
```

Grafico Esercizio 3 - Esercitazione 3



```
% Esercizio 4 - Esercitazione 3
clear all
close all
clc
format short


x=1;
val_esa=exp(x);

alfa=[1:1:20];
h=10.^(-alfa)

appr1 = (exp(x+h)-exp(x))./h;
appr2 = (exp(x+h)-exp(x-h))./(2.*h);
err1 = val_esa - appr1;
err2 = val_esa - appr2;

% Tabella in formato short e che contiene i valori h e i corrispondenti
% errori analitici.
format short e
risultati= [h' err1' err2']
% dalla tabella dei risultati notiamo che al decrescere del parametro h i
% valori dei due errori si avvicinano.

format short
```

COMMAND WINDOW

```
h =
    Columns 1 through 11
        0.1000    0.0100    0.0010    0.0001    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
   Columns 12 through 20
        0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
risultati =
        1.0000e-01  -1.4056e-01  -4.5327e-03
        1.0000e-02  -1.3637e-02  -4.5305e-05
        1.0000e-03  -1.3596e-03  -4.5305e-07
        1.0000e-04  -1.3592e-04  -4.5306e-09
        1.0000e-05  -1.3591e-05  -5.8587e-11
        1.0000e-06  -1.3585e-06   1.6346e-10
        1.0000e-07  -1.3551e-07  -5.8587e-11
        1.0000e-08   5.1012e-08   6.6028e-09
        1.0000e-09   2.2865e-07   6.6028e-09
        1.0000e-10   2.8932e-06   6.7274e-07
        1.0000e-11   1.1775e-05  -1.0429e-05
        1.0000e-12   1.1775e-05  -2.1027e-04
        1.0000e-13   4.8968e-03   4.5586e-04
        1.0000e-14   5.3747e-02   9.3376e-03
        1.0000e-15   5.3747e-02  -1.6830e-01
        1.0000e-16   2.7183e+00   4.9784e-01
        1.0000e-17   2.7183e+00   2.7183e+00
        1.0000e-18   2.7183e+00   2.7183e+00
        1.0000e-19   2.7183e+00   2.7183e+00
        1.0000e-20   2.7183e+00   2.7183e+00
```

```
% Esercizio 5 - Esercitazione 3
clear all
close all
clc
format long
% Matrice di Hilbert
n=10;
H=hilb(n)
% b è il vettore dei termini noti dato dal prodotto della matrice di
% Hilbert e il vettore delle incognite con tutti gli elementi posti
% uguali a 1
x=ones(10,1)
b= H*x
% risoluzione sistema lineare tramite comandi matlab
x = H\b
% il vettore x è perturnato da errori perche H è mal condizionata.
% costruiamo il vettore c secondo il testo dell'esercizio
z=[0.001 zeros(1,9)];
% il vettore z è il vettore perturbazione
c = b + z'
% risolviamo il sistema con c come vettore dei termini noti, che è formato
% dal vettore b piu il vettore z, quindi il vettore c è perturbato.
y = H\c
% calcoliamo l'errore relativo, per analizzare la perturbazione di y
er=norm(x-y)/norm(x)
format short
```

COMMAND WINDOW

```
H =

  Columns 1 through 6

  1.000000000000000   0.500000000000000   0.333333333333333   0.250000000000000   0.200000000000000
0.166666666666667
  0.500000000000000   0.333333333333333   0.250000000000000   0.200000000000000   0.166666666666667
0.142857142857143
  0.333333333333333   0.250000000000000   0.200000000000000   0.166666666666667   0.142857142857143
0.125000000000000
  0.250000000000000   0.200000000000000   0.166666666666667   0.142857142857143   0.125000000000000
0.111111111111111
  0.200000000000000   0.166666666666667   0.142857142857143   0.125000000000000   0.111111111111111
0.100000000000000
  0.166666666666667   0.142857142857143   0.125000000000000   0.111111111111111   0.100000000000000
0.090909090909091
  0.142857142857143   0.125000000000000   0.111111111111111   0.100000000000000   0.090909090909091
0.083333333333333
  0.125000000000000   0.111111111111111   0.100000000000000   0.090909090909091   0.083333333333333
0.076923076923077
  0.111111111111111   0.100000000000000   0.090909090909091   0.083333333333333   0.076923076923077
0.071428571428571
  0.100000000000000   0.090909090909091   0.083333333333333   0.076923076923077   0.071428571428571
0.066666666666667
```

Columns 7 through 10

0.142857142857143	0.125000000000000	0.111111111111111	0.100000000000000
0.125000000000000	0.111111111111111	0.100000000000000	0.090909090909091
0.111111111111111	0.100000000000000	0.090909090909091	0.083333333333333
0.100000000000000	0.090909090909091	0.083333333333333	0.076923076923077
0.090909090909091	0.083333333333333	0.076923076923077	0.071428571428571
0.083333333333333	0.076923076923077	0.071428571428571	0.066666666666667
0.076923076923077	0.071428571428571	0.066666666666667	0.062500000000000
0.071428571428571	0.066666666666667	0.062500000000000	0.058823529411765
0.066666666666667	0.062500000000000	0.058823529411765	0.055555555555556
0.062500000000000	0.058823529411765	0.055555555555556	0.052631578947368

x =

1
1
1
1
1
1
1
1
1
1

b =

2.928968253968254

2.019877344877345
1.603210678210678
1.346800421800422
1.168228993228993
1.034895659895660
0.930728993228993
0.846695379783615
0.777250935339171
0.718771403175428

x =

0.999999999885724
1.000000009323761
0.999999810312795
1.000001659597570
0.999992339982267
1.000020461923272
0.999967268915579
1.000030922478050
0.999984093616664
1.000003434004451

c =

2.929968253968254
2.019877344877345
1.603210678210678
1.346800421800422
1.168228993228993
1.034895659895660
0.930728993228993
0.846695379783615
0.777250935339171
0.718771403175428

y =

1.0e+03 *

0.001099996568998
-0.003949699846959
0.080193547416395
-0.599540918318498
2.523236585573656
-6.304517292276935
9.609310908514386
-8.749350248055125
4.376142102142063
-0.922634979112010

er =

4.851509154407465e+03