APPELLO di ANALISI NUMERICA

Dott. S. De Marchi

`Verona,...............`

Il candidato dovrà fornire il floppy contenente i files dei programmi e scrivere il proprio nome e matricola sull'etichetta.

1 Si consideri la funzione $f(x) = \sin(3x) + x^2$ nell'intervallo $[a, b] = [1, \pi]$. Si costruisca poi una partizione $\Delta_N = \{a = x_1 < x_2 < \cdots < x_N = b\}$ di [a,b] sulla quale determinare il polinomio a tratti cubico interpolante $f(x)$. Per la costruzione di detto polinomio procedere come segue.

Su ogni intervallino $I_k = [x_k, x_{k+1}]$, $k = 1, ..., N-1$ si costruisca *l'interpolante cubica di Hermite*, ovvero il polinomio cubico $H_{3,k}$ tale che:

$$\begin{array}{ll} H_{3,k}(x_k) = f(x_k); & H_{3,k}(x_{k+1}) = f(x_{k+1}); \\ H'_{3,k}(x_k) = f'(x_k); & H'_{3,k}(x_{k+1}) = f'(x_{k+1}); \end{array} \tag{1}$$

Si chiede di scrivere un programma MATLAB che determina per ogni intervallino l'interpolante di Hermite a tratti cubica. Fare anche i grafici e determinare in norma 2 l'errore tra funzione e interpolante. Commentare i risultati.

2 Dato il polinomio

$$p_n(x) = \sum_{i=0}^{n} a_i x^i, \quad a_0 a_n \neq 0 \ ,$$

per determinare la radice ξ_1 di modulo massimo si può usare il *metodo di Bernoulli*. Tale metodo consiste nell'applicare il *metodo delle potenze* alla matrice (di Frobenius)

$$F = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ & \ddots & \ddots & & \\ & & \ddots & \ddots & \\ 0 & \ldots & & 0 & 1 \\ -\frac{a_0}{a_n} & -\frac{a_1}{a_n} & \ldots & -\frac{a_{n-2}}{a_n} & -\frac{a_{n-1}}{a_n} \end{pmatrix}$$

1. Calcolare ξ_1 per il polinomio

$$p_6(x) = 13x^6 - 364x^5 + 2912x^4 - 9984x^3 + 16640x^2 - 13312x + 4096 \ .$$

2. (*Facoltativo*) Calcolare quindi per deflazione ξ_2, cioè la seconda radice. Si suggerisce di costruire la matrice P di Householder tale

$$PFP^T = \begin{pmatrix} \xi_1 & \mathbf{a}^T \\ \mathbf{0} & F_1 \end{pmatrix}$$

cosicchè $Px_1 = e_1$ con x_1 autovettore associato a ξ_1 (calcolato al passo 1) e $e_1 = (1, 0, ..., 0)^T$. Per calcolare ξ_2 applicare il metodo delle potenze alla matrice F_1.

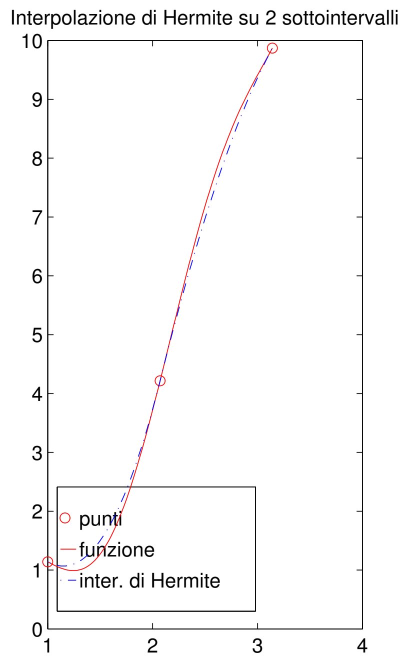
Confrontare i risultati con la funzione `roots(c)`, con `c` vettore dei coefficienti del polinomio $p_6(x)$.

◇◇

Chi si presenta per il *I modulo* o il *II modulo* deve risolvere il primo o il secondo esercizio in 1 ora e mezza.
Chi si presenta per il *I e II modulo* deve risolvere i due esercizi in 2 ore e mezza.

```
%------------  SOLUZIONI ----------------



%----------------------------------------
% 1^ Esercizio
%-----------------------------------------

clear;
a1=input('Dammi a = ');
b1=input('Dammi b = ');
m=input('Numero di sottointervalli = ');

h1=(b1-a1)/m;
xi=a1:h1:b1;
n=length(xi);

for k=1:n-1,

  a=xi(k); b=xi(k+1);
  [fa(k),dfa]=ff(a);
  [fa(k+1),dfb]=ff(b);

 V=[1 a a^2 a^3
    0 1 2*a 3*a^2
    1 b b^2 b^3
    0 1 2*b 3*b^2];

 d=[fa(k);dfa;fa(k+1);dfb];

 c(:,k)=inv(V)*d;

end;


%---------- PLOTS --------

h=(b1-a1)/100;
x=a1:h:b1;

for i=1:length(x),
```

```
   for k=1:n-1,
      if(x(i)>=xi(k) & x(i)<=xi(k+1))
         l=k;
         break;
      end;
   end;

   [f1(i),f1d(i)]=ff(x(i));
   p(i)=c(4,l);
   for j=3:-1:1,
      p(i)=c(j,l)+p(i)*x(i);
   end;
end;
% Errore in norma 2
err=norm(f1-p,2)

plot(xi,fa,'ro',x,f1,'r-',x,p,'b-.');
legend('punti','funzione','inter. di Hermite');
s_tit=['Interpolazione di Hermite su ',num2str(m),' sottointervalli'];
title(s_tit);
print -depsc comp33.eps


%-------------------------
function [f1,df1]=ff(x)
f1=sin(3*x)+x^2;
df1=cos(3*x)*3+2*x;
return;


%------------- RISULTATI
>> comp33

Dammi a = 1

Dammi b = pi

Numero di sottointervalli = 5

err =

    0.0307
```




```
>> comp33

Dammi a = 1

Dammi b = pi

Numero di sottointervalli = 10

err =

    0.0020
```

Commento.
Come si vede l'approssimazione migliora al crescere del numero dei sottointervalli (come deve essere se la funzione e' sufficientemente regolare, come si deduce dai teoremi visti).

```
%-------------------------------------
% 2^ Esercizio
%-------------------------------------
clear;
nmax=1000;
tol=1.e-6;

c=[4096 -13312 16640 -9984 2912 -364 13];

c1=c(1:length(c)-1)/c(length(c)); %normalizzo
n=length(c1);

% Costruisco la matrice di Frobenius

F(n,:)=-c1;
F=F+diag(ones(1,n-1),1);

% Applico il metodo delle potenze per il calcolo
% dell'autovalore di modulo max della matrice F
t0=ones(n,1);
m0=1000;
t1=F*t0;
m1=max(abs(t1));
t1=t1/m1;
k=1;

while( abs(m1-m0) >tol & k <= nmax  )
   t0=t1;
   m0=m1;
   t1=F*t0;
   m1=max(abs(t1));
   t1=t1/m1;
   k=k+1;
end;
rmax1=m1
t1

% Calcolo del secondo autovalore-radice per deflazione
% Costruisco la matrice di Householder P t.c. P*t1=(1,0,...,0)
t12=norm(t1,2);
beta=1/(t12*(t12+abs(t1(1)) ));
```

```
v(1)=sign(t1(1))*(abs(t1(1))+t12);
v(2:n)=t1(2:n);
P=eye(n)-beta*v'*v;

F1=P*F*P';

%Per determinare il successivo autovalore devo riapplicare il metodo delle
% potenze alla sottomatrice di F1 di dimensione n-1 x n-1

F2=F1(2:n,2:n);

t0=ones(n-1,1);
m0=1000;
t1=F2*t0;
m1=max(abs(t1));
t1=t1/m1;
k=1;

while( abs(m1-m0) >tol & k <= nmax  )
   t0=t1;
   m0=m1;
   t1=F2*t0;
   m1=max(abs(t1));
   t1=t1/m1;
   k=k+1;
end;
rmax2=m1
t2=t1;
t2


%---------------- RISULTATI

rmax1 =

   17.4605


t1 =

   -0.0000
   -0.0000
```

```
   -0.0002
   -0.0033
   -0.0573
   -1.0000


rmax2 =

    4.6303

t2 =

   -0.0137
   -0.0625
   -0.2740
   -1.0000
    0.0612

>> eig(F)

ans =

   17.4605
    4.6303
    2.2741
    1.4764
    1.1438
    1.0147
```