```
close all

%%%%%%%%%%%%%%%%%%%%%
%  Esercizio 1
%%%%%%%%%%%%%%%%%%%%%
fprintf('Approssimazione  spline per funzione sin(x)cos(x) \n')

f=@(x)(sin(x).*cos(x));
alfa=0; beta=2*pi;

%n.1
figure(1)
fplot(f,[alfa,beta],'r--');
hold on

%n.2
t=linspace(alfa,beta,10000)';
for n=[4 8 12]

    x = linspace(alfa,beta,n+1)';
    s = spline(x,f(x),t);     % not-a-knot
    plot(t,s);
   %pause

end
hold off
title('approssimazione spline not-a-knot')
legend('f','spline 4','spline 8','spline 12')

%n.3
figure(2)
fplot(f,[alfa,beta],'r--');
hold on

t=linspace(alfa,beta,10000)';
d1=1; d2=1;
for n=[4 8 12]

    x = linspace(alfa,beta,n+1)';
    s = spline(x,[d1;f(x);d2],t);     % completa
    plot(t,s);
   %pause

end
hold off
title('approssimazione spline completa d1=1,d2=1')
legend('f','spline 4','spline 8','spline 12')

figure(3)
fplot(f,[alfa,beta],'r--');
hold on

t=linspace(alfa,beta,10000)';
d1=-1; d2=-1;
for n=[4 8 12]

    x = linspace(alfa,beta,n+1)';
    s = spline(x,[d1;f(x);d2],t);     % completa
    plot(t,s);
   %pause

end
hold off
title('approssimazione spline completa d1=-1,d2=-1')
legend('f','spline 4','spline 8','spline 12')
```

```
%%%%%%%%%%%%%%%%%%%%%
%  Esercizio 2
%%%%%%%%%%%%%%%%%%%%%
fprintf('Convergenza spline per funzione di Runge\n')


f=@(x)(1./(1+x.^2));
alfa=-5;  beta=5;

figure(10)
fplot(f,[alfa,beta],'r--');
hold on

%n.1

%t=linspace(alfa,beta,10000)';
%for n=[6 8 10]
%    x = linspace(alfa,beta,n+1)';
%    s = spline(x,f(x),t);     % not-a-knot
%    plot(t,s);
%end
%hold off
%title('approssimazione spline not-a-knot')
%legend('f','spline 6','spline 8','spline 10')

t=linspace(alfa,beta,1000000)';
fprintf('       k           Errore         ordine \n')
for k=1:10
    x=(alfa:(beta-alfa)/2^k:beta)';
    s=spline(x,f(x),t);
    Er(k)=max(abs(f(t)-s));
    if k>1
      p=(1/log(1/2))*log(Er(k)/Er(k-1));
      disp([k,Er(k),p]);
    end
end


%n.2
fprintf(' Grafico spline e derivate \n')

n=10;
x=linspace(alfa,beta,n+1);
S=spline(x,f(x));

d1f=@(x)(-2*x./(1+x.^2).^2);
d2f=@(x)(-2./(1+x.^2).^2+8*x.^2./(1+x.^2).^3);
d3f=@(x)(24*x./(1+x.^2).^3 - 48*x.^3./(1+x.^2).^4);

figure(100)
subplot(1,4,1); fplot(f,[alfa,beta],'-'); hold on
subplot(1,4,2); fplot(d1f,[alfa,beta],'-'); hold on
subplot(1,4,3); fplot(d2f,[alfa,beta],'-'); hold on
subplot(1,4,4); fplot(d3f,[alfa,beta],'-'); hold on

rho=S.coefs;
drho=[3*rho(:,1),  2*rho(:,2), rho(:,3)];
d2rho=[6*rho(:,1), 2*rho(:,2)];
d3rho=6*rho(:,1);

                    Sval=ppval(S,t);        % valutazione spline
dS=mkpp(x,drho);    dSval=ppval(dS,t);      % valutazione der I spline
d2S=mkpp(x,d2rho);  d2Sval=ppval(d2S,t);    % valutazione der II spline
d3S=mkpp(x,d3rho);  d3Sval=ppval(d3S,t);    % valutazione der III spline
```

```
subplot(1,4,1); plot(t,Sval,'--');hold off     % plot spline
subplot(1,4,2); plot(t,dSval,'--');hold off    % plot der I spline
subplot(1,4,3); plot(t,d2Sval,'--');hold off   % plot der II spline
subplot(1,4,4); plot(t,d3Sval,'--');hold off   % plot der III spline


% Es.3
fprintf(' Studio ordine di convergenza di tutte le derivate\n')
t=linspace(alfa,beta,1000000)';
i=0;
fprintf('       k          Er f         ordine        Er df       ordine      Er 
d2f       ordine       Er d3f       ordine \n')
for n=2:10,
   i=i+1;
   x=(alfa:1/2^n:beta)';
   S=spline(x,f(x));
   rho=S.coefs;
   drho=[3*rho(:,1) 2*rho(:,2) rho(:,3)]; d1S=mkpp(x,drho);
   d2rho=[6*rho(:,1)  2*rho(:,2)];        d2S=mkpp(x,d2rho);
   d3rho=6*rho(:,1);                      d3S=mkpp(x,d3rho);

   Er0(i)=max(abs(ppval(S,t)-f(t)));
   Er1(i)=max(abs(ppval(d1S,t)-d1f(t)));
   Er2(i)=max(abs(ppval(d2S,t)-d2f(t)));
   Er3(i)=max(abs(ppval(d3S,t)-d3f(t)));

   if (i>1),
      c0=1/log(1/2)*log(Er0(i)/Er0(i-1));
      c1=1/log(1/2)*log(Er1(i)/Er1(i-1));
      c2=1/log(1/2)*log(Er2(i)/Er2(i-1));
      c3=1/log(1/2)*log(Er3(i)/Er3(i-1));
      disp([i,Er0(i),c0,Er1(i),c1,Er2(i),c2,Er3(i),c3]),
   end

end









%%%%%%%%%%%%%%%%%%%%%%%%%%
%  Esercizio 3  (Facoltativo)
%%%%%%%%%%%%%%%%%%%%%%%%%%
warning off

dati=[65 3.52
55 3.62
45 3.65
35 3.52
25 3.47
15 3.25
 5 3.15
-5 3.15
-15 3.2
-25 3.27
-35 3.52
-55 3.7];

dato_nuovo=[-45 3.7];
```

```
%n.1
plot(dati(:,1),dati(:,2),'*');hold on
t=linspace(-70,80,10000)';
s=spline(dati(:,1),dati(:,2),t);
plot(t,s);

%n.2
a=get_polyn(dati(:,1),dati(:,2));
plot(t,polyval(a,t),'--');
axis([-70,80,2,5])
legend('dati','spline','lagrange')
title('approssimazione dati temperatura')
hold off

%n.3
x_new=-45;
y_new=3.7;
ylagrange=polyval(a,x_new);
yspline=spline(dati(:,1),dati(:,2),x_new);
% In alternativa, salvare prima la struttura S, e poi valutare S con ppval in x_new


fprintf('Stima di ynew. Val esatto: %d, val.Lagrange: %d, val. spline:
%d\n',y_new,ylagrange,yspline)
```