```
% EDO de dimensión 1 simple con resolución con codigo propio
clear all
close all
f=@(t,u) cos(t*u)*sin(t) ; % Función de la ED
t0=2;       % Tiempo inicial
tf=50;      % Tiempo final
solu(1)=1;  % Solucion inicial
dt=0.01;     %Paso de tiempo
tline=t0:dt:tf; %Los tiempos que simulamos

for i=2:length(tline)
    solu(i)=solu(i-1)+dt*f(t0+dt*i,solu(i-1));
end

figure(1)
clf
plot(tline,solu,'linewidth',2)
grid on
axis tight
title('Solution')
xlabel('Time (s)')
ylabel('u(t)')


% EDO con el resolvedor de Matlab
[t,y] = ode45(f, [2 50], 1);

figure(2)
clf
plot(t,y,'linewidth',2)
grid on
axis tight
title('Solution')
xlabel('Time (s)')
ylabel('u(t)')

% PARA HACER: CON EULER Y ODE45
% Resolver du/dt=t-log(u) con u(1)=1 y estimar u(100)



% EDO de dimensión 1 acopladas
clear all

f=@(u,v,t) -0.1*v*u+0.1*u; % Función de la ED
g=@(u,v,t) +0.1*u*v-0.5*v;


t0=0;       % Tiempo inicial
tf=100;      % Tiempo final
solu(1)=0.5;  % Solucion inicial
solv(1)=0.7;  % Solucion inicial
dt=0.01;     %Paso de tiempo
tline=t0:dt:tf; %Los tiempos que simulamos
```

```
for i=2:length(tline)
    solu(i)=solu(i-1)+dt*f(solu(i-1),solv(i-1),t0+dt*i);
    solv(i)=solv(i-1)+dt*g(solu(i-1),solv(i-1),t0+dt*i);
end
figure(3)
clf
hold on
plot(tline,solu,'b','linewidth',2)
plot(tline,solv,'r','linewidth',2)
grid on
axis tight
title('Solution')
xlabel('Time (s)')
ylabel('Density')
legend('Subs.','Bact.')


h=@(t,u) [-0.1*u(1)*u(2)+0.1*u(1);+0.1*u(1)*u(2)-0.5*u(2)];

[t,y] = ode45(h,[0 100],[0.5; 0.7]);
figure(4)
clf
hold on
plot(t,y(:,1),'b','linewidth',2)
plot(t,y(:,2),'r','linewidth',2)
grid on
axis tight
title('Solution')
xlabel('Time (s)')
ylabel('Density')
legend('Subs.','Bact.')

% PARA HACER: CON EULER Y ODE45
% du/dt=v   y   dv/dt=0.1*(1-u^2)*v-v; con u(0)=0.4 y v(0)=0.6 y estimar la solucion a
tiempo 30
% Con cambio de variables resolver d^2u/dt^2+u*du/dt= 0 con u(0)=0.2 y u/dt(0)=0.4 y
estimar la solucion a tiempo 10
```

```
clear all
close all
%Creation del tiempo fisico
h=0.001; %paso de tiempo
I=0:h:9;
%%%%%% Definicion de las condiciones iniciales
xs(1)=1; %% corresponde a la soluicon en el tiempo 0
xs(2)=1;
%%%%% Iteration de Taylor orden 2
for n=3:length(I)
    xs(n)=2*xs(n-1)-xs(n-2)+h^2*((4*I(n)^2-3)*...
        exp(-I(n)^2)+xs(n-1));
end

%%%%% Metodo de resolucion mediante sistema de
%%%%% ecuacion de orden 1
xs1(1)=1;  %% esta variable es la solucion de mi sistema
xs2(1)=0;
%%%% Metodo de Taylor orden 1
for n=2:length(I)
     xs1(n)=xs1(n-1)+h*(xs2(n-1));
     xs2(n)=xs2(n-1)+h*(xs1(n-1)+((4*I(n)^2-3)*exp(-I(n)^2)));
end


%%%%% Solucion exacta:
 for n=1:length(I)
     xe(n)=exp(-I(n)^2);
 end

%%%% calcular el error
for n=1:length(I)
 errxs(n)=abs(xs(n)-xe(n));
 errxs1(n)=abs(xs1(n)-xe(n));
end
disp(['error medio metodo T2:' num2str(mean(errxs))])
disp(['error medio metodo ST1:' num2str(mean(errxs1))])

figure(1)
clf
hold on
plot(I,xs,'b')
plot(I,xs1,'g')
plot(I,xe,'r')
legend('T2','ST1','Ex.')
xlabel('Time')
ylabel('Solution')

figure(2)
clf
hold on
plot(I,errxs,'b')
plot(I,errxs1,'g')
legend('T2','ST1')
xlabel('Time')
```

```
ylabel('Error')
```

```
clear all
close all
eqc=1;
eqo=1;

dt=.01;
dx=.1;
Itmax=1000;
L=0:dx:5;

if eqc==1
%%%%% EDP: Ecuacion de calor
%%%%% k d2u/dx2= du/dt, U(t,0)=u(t,L)=0, u(x,0)=f(x)

k=.1; %.01 .5 1.
eps=.01;
%%%%% Estado inicial f(x)
t=1;
uc=zeros(Itmax,length(L));
uc(t,20:30)=1;
%%%%% Output grafico
figure(1)
plot(L,uc(t,:),'r')

%%%%% Bucle explicita
while(t<Itmax)&(norm(uc(t,:),2)>eps)
    t=t+1;
for x=2:length(L)-1
  uc(t,x)=uc(t-1,x)+dt*(k*(uc(t-1,x+1)+uc(t-1,x-1)-2*uc(t-1,x))/(dx^2));
end

%%%%% Output grafico
figure(1)
clf
plot(L,uc(t,:),'r')
axis([0 5 0 1])
title(['Solution | Current norm: ' num2str(norm(uc(t,:),2)) ' | iteration: ' num2str↙
(t) ])
end

figure(2)
clf
surface(uc,'edgecolor','none','facecolor','interp')
colorbar
ylabel('tiempo')
xlabel('espacio')
zlabel('Temperature')
axis tight
view(45,45)
end

if eqo==1
%%%%% EDP: Ecuacion de ondas
%%%%% a^2 d2u/dx2= d2u/dt2, U(t,0)=u(t,L)=0, u(x,0)=f(x), u'(x,0)=g(x)
a=1; %.01 .5 1.
```

```
eps=.01;
%%%%% Estado inicial f(x)
t=1;
uo=zeros(Itmax,length(L));
uo(t,:)=0;
g=-0.5*cos(2*pi*L/5);
t=t+1;
uo(t,:)=uo(t,:)+g;
%%%%% Output grafico
figure(1)
plot(L,uo(t,:),'r')

%%%%% Buole explicita
while(t<Itmax)&(norm(uo(t,:),2)>eps)
    t=t+1;
for x=2:length(L)-1
  uo(t,x)=-uo(t-2,x)+2*uo(t-1,x)+(dt^2)*(a^2*(uo(t-1,x+1)+uo(t-1,x-1)-2*uo(t-1,x))/
(dx^2));
end

%%%%% Output grafico
figure(1)
clf
plot(L,uo(t,:),'r')
axis([0 5 -100 100])
title(['Solution | Current norm: ' num2str(norm(uo(t,:),2)) ' | iteration: ' num2str
(t) ])
end

figure(2)
clf
surface(uo,'edgecolor','none','facecolor','interp')
colorbar
ylabel('tiempo')
xlabel('espacio')
zlabel('Temperature')
axis tight
view(45,45)
end
```