

**ITESO**

**MÉTODOS NUMÉRICOS**

**LÓPEZ LAZARENO DIEGO ALBERTO IF722100**

**PRÁCTICA 10**

1. Realiza un programa en Matlab que resuelva EDO mediante el método de Heun y RK-4.

% Ecuación diferencial ordinaria

function ode=ode(x,y)

ode=3\*y\*(x^3)-5\*y;

end

% Método de Euler

x=input("Ingrese el valor inicial de x");

y=input("Ingrese el valor inicial de y");

h=input("Ingrese el tamaño del paso h");

xl=input("Ingrese el valor final de x");

% Iteraciones

n=(xl-x)/h;

% Ciclo for para estimar la curva solución de la EDO

for i=1:n

% Pendiente

k=ode(x(i),y(i));

% x+h

x(i+1)=x(i)+h;

% Estimación de y

y(i+1)=y(i)+k\*h;

end

% Solución analítica

y\_analitica=1;

for i=1:length(x)

y\_analitica(i)=9.9\*exp(0.75\*(x(i)^4)-5\*x(i));

end

% Visualización

% Visualización

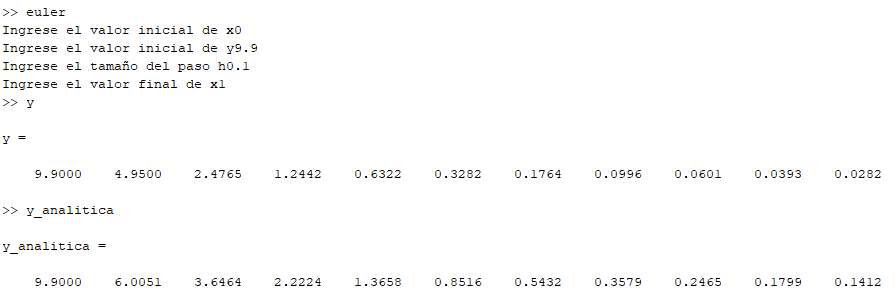
plot(x,y)

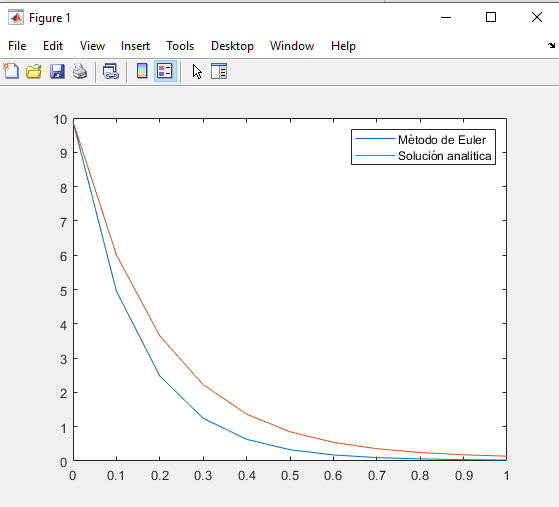
hold on

plot(x,y\_analitica)

hold off

legend({'Método de Euler','Solución analitica'})





Solución analítica y solución aproximada por el método de Euler

% Método de Heun

x=input("Ingrese el valor inicial de x");

y=input("Ingrese el valor inicial de y");

h=input("Ingrese el tamaño del paso h");

xl=input("Ingrese el valor final de x");

% Iteraciones

n=(xl-x)/h;

% Ciclo for para estimar la curva solución de la EDO

for i=1:n

% Pendiente k1

k1=ode(x(i),y(i));

% Pendiente k2

k2=ode(x(i)+h,y(i)+k1\*h);

% x+h

x(i+1)=x(i)+h;

% Estimación de y

y(i+1)=y(i)+(k1+k2)\*h/2;

end

% Solución analítica

y\_analitica=1;

for i=1:length(x)

y\_analitica(i)=9.9\*exp(0.75\*(x(i)^4)-5\*x(i));

end

% Visualización

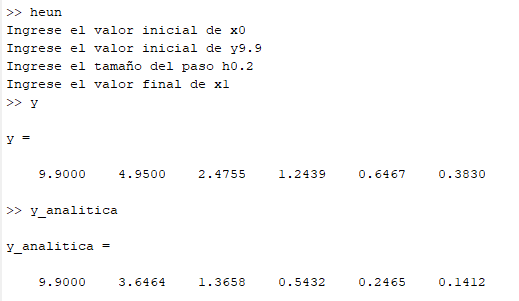
plot(x,y)

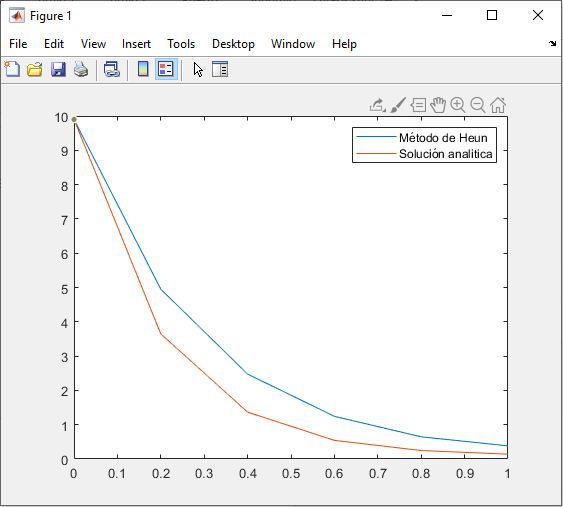
hold on

plot(x,y\_analitica)

hold off

legend({'Método de Heun','Solución analitica'})





Solución analítica y solución aproximada por el método de Heun

% Método de Runge-Kutta 4to orden

x=input("Ingrese el valor inicial de x");

y=input("Ingrese el valor inicial de y");

h=input("Ingrese el tamaño del paso h");

xl=input("Ingrese el valor final de x");

% Iteraciones

n=(xl-x)/h;

% Ciclo for para estimar la curva solución de la EDO

for i=1:n

% Pendiente k1

k1=ode(x(i),y(i));

% Pendiente k2

k2=ode(x(i)+0.5\*h,y(i)+0.5\*k1\*h);

% Pendiente k3

k3=ode(x(i)+0.5\*h,y(i)+0.5\*k2\*h);

% Pendiente k4

k4=ode(x(i)+h,y(i)+k3\*h);

% x+h

x(i+1)=x(i)+h;

% Estimación de y

y(i+1)=y(i)+(k1+2\*k2+2\*k3+k4)\*h/6;

end

% Solución analítica

y\_analitica=1;

for i=1:length(x)

y\_analitica(i)=9.9\*exp(0.75\*(x(i)^4)-5\*x(i));

end

% Visualización

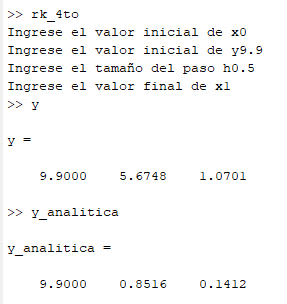
plot(x,y)

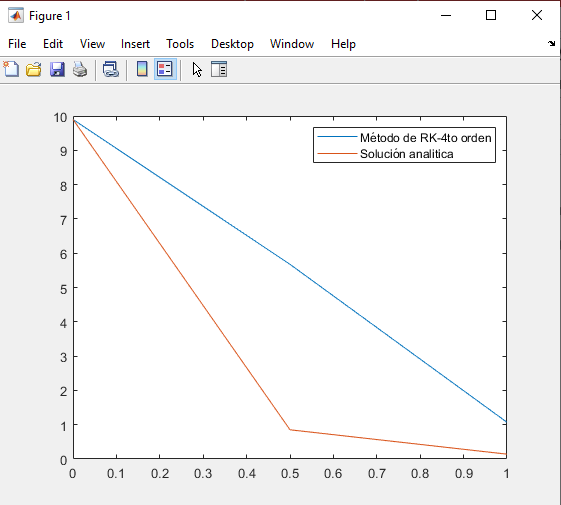
hold on

plot(x,y\_analitica)

hold off

legend({'Método de RK-4to orden','Solución analitica'})





Solución analítica y solución aproximada por el método de RK-4to orden

1. Investiga cómo utilizar el comando ode45

[t,y] = ode45(odefun,tspan,y0)

[t,y] = ode45(odefun,tspan,y0,options)

[t,y,te,ye,ie] = ode45(odefun,tspan,y0,options)

sol = ode45(\_\_\_)

[[t](https://la.mathworks.com/help/matlab/ref/ode45.html" \l "bu00_4l_sep_shared-t),[y](https://la.mathworks.com/help/matlab/ref/ode45.html#bu00_4l_sep_shared-y)] = ode45([odefun](https://la.mathworks.com/help/matlab/ref/ode45.html#bu00_4l_sep_shared-odefun),[tspan](https://la.mathworks.com/help/matlab/ref/ode45.html#bu00_4l_sep_shared-tspan),[y0](https://la.mathworks.com/help/matlab/ref/ode45.html#bu00_4l_sep_shared-y0)), donde tspan = [t0 tf] integra el sistema de ecuaciones diferenciales *y*′=*f*(*t*,*y*) de t0 a tf con condiciones iniciales y0. Cada fila del array de soluciones y corresponde a un valor devuelto en el vector de columna t.

**Conclusión**

En esta práctica de laboratorio se abordó la solución de una ecuación diferencial ordinaria de primer orden con valor inicial mediante el Método de Euler, el Método de Heun y el Método de Runge-Kutta de 4to orden. Para este ejercicio en particular la aproximación que mejor resultó fue la obtenida a través del Método de Heun. Por otra parte, queda claro que la curva obtenida por cualquier método estará más cercana a la curva obtenida analíticamente si dejamos que nuestros incrementos en “h” estén cercanos al 0.