PROBLEMA 1: STURM-LIOUVILLE

```
clearvars
clear all
clc
close all
```

Problema 1

En los siguientes problemas de Sturm-Liouville definidos con la ecuación diferencial

$$\frac{d^2 y}{d x^2} + \lambda^2 y = 0, \quad 0 < x < 1$$

y cada una de las siguientes combinaciones de condiciones de frontera

a) $y(a) = y(b) = 0$

b) $y(a) = y'(b) = 0$

c) $y'(a) = y'(b) = 0$

d) $y'(a) = \beta_1 y(a) \quad y \quad -y'(b) = \beta_2 y(b)$

i) Encuentre la función propia, la condición de los valores propios, los valores propios y la condición de ortogonalidad.

ii) Para el caso en que $a = 0$ y $b = 1$, muestre gráficamente las funciones correspondientes al primer, quinto, vigésimo y quincuagésimo valor propio.

Para el inciso (d) desarrolle el algebra en forma general, pero los cálculos hágalos solo para cuando $\beta_1 = 0.01$ y $\beta_2 = 1.0$.

SOLUCIÓN:

Se identifica, de la ecuación diferencial que propone el problema, la ecuación del movimiento armónico simple. Recordamos que su solución viene dada por:

$$y = A\cos(\lambda x + \phi)$$

Donde A y ϕ son parámetros reales que dependen de las condiciones iniciales del problema:

i) Función propia, condición de valores propios, valores propios y condición de ortogonalidad:

Función propia:

Para las condiciones a), b),c) y d), se tiene una misma función propia. Ésta es la ecuación $y = y(\lambda, x)$ mostrada anteriormente.

Condición de valores propios:

Basta con evaluar lo anterior en las condiciones a), b),c) y d).

Condición a):

$$y(a) = A\cos(\lambda a + \phi) = 0$$
$$y(b) = A\cos(\lambda b + \phi) = 0$$

Que por tanto:

$$\lambda a + \phi = (\frac{2n + 1}{2})\pi$$
$$\lambda b + \phi = (\frac{2m + 1}{2})\pi$$

Son las condiciones de valores propios, con "n" y "m" números enteros.

Condición b):

$$y(a) = A\cos(\lambda a + \phi) = 0$$
$$y'(b) = -A\lambda\sin(\lambda b + \phi) = 0$$

Que por tanto:

$$\lambda a + \phi = (\frac{2n + 1}{2})\pi$$
$$\lambda b + \phi = m\pi$$

Son las condiciones de valores propios, con "n" y "m" números enteros.

Condición c):

$$y'(a) = -A\lambda\sin(\lambda a + \phi) = 0$$
$$y'(b) = -A\lambda\sin(\lambda b + \phi) = 0$$

Que por tanto:

$$\lambda a + \phi = n\pi$$
$$\lambda b + \phi = m\pi$$

Son las condiciones de valores propios, con "n" y "m" números enteros.

Condición d):

$$y'(a) = -A\lambda\sin(\lambda a + \phi) = \beta_1 y(a) = \beta_1 A\cos(\lambda a + \phi)$$
$$-y'(b) = A\lambda\sin(\lambda b + \phi) = \beta_2 y(b) = \beta_2 A\cos(\lambda b + \phi)$$

Que por tanto:

$$\beta_1\cos(\lambda a + \phi) + \lambda\sin(\lambda a + \phi) = 0$$
$$\beta_2\cos(\lambda b + \phi) + \lambda\sin(\lambda b + \phi) = 0$$

Son las condiciones de valores propios.

Valores propios:

Condición a):

Del sistema de ecuaciones, se encuentra:

$$\lambda = (\frac{n - m}{a - b})\pi = \lambda_{nm}$$

Como los valores propios.

Condición b):

Del sistema de ecuaciones, se encuentra:

$$\lambda = \left(\frac{2(n-m)+1}{2}\right)\pi = \lambda_{nm}$$

Como los valores propios.

Condición c):

Del sistema de ecuaciones, se encuentra:

$$\lambda = \left(\frac{n-m}{a-b}\right)\pi = \lambda_{nm}$$

Como los valores propios.

Condición d):

Del sistema de ecuaciones, se encuentra:

$$\sin(\mu_1)\cos(\lambda a+\phi)+\cos(\mu_1)\sin(\lambda a+\phi) = 0$$
$$\sin(\mu_2)\cos(\lambda b+\phi)+\cos(\mu_2)\sin(\lambda b+\phi) = 0$$

Es decir:

$$\sin(\mu_1+\lambda a+\phi) = 0$$
$$\sin(\mu_2+\lambda b+\phi) = 0$$

Esto es:

$$\mu_1+\lambda a+\phi = n\pi$$
$$\mu_2+\lambda b+\phi = m\pi$$

Finalmente:

$$\lambda = \frac{(n-m)\pi + \mu_2 - \mu_1}{(a-b)}$$

Como los valores propios. Donde:

$$\sin(\mu_1) = \frac{\beta_1}{\sqrt{(\lambda^2+\beta_1^2)}}$$
$$\sin(\mu_2) = \frac{\beta_2}{\sqrt{(\lambda^2+\beta_2^2)}}$$

Condición de ortogonalidad:

Por teoría, valores propios diferentes corresponden a Función propias ortogonales. Ahora, si:

$$y_{nm} = y(\lambda_{nm}, x) = A\cos(\lambda_{nm}x+\phi)$$

Luego:

$$\int_a^b y_{\mathrm{nm}} y_{\mathrm{pq}} \mathrm{dx} = \delta_{(n-m)(p-q)} \int_a^b y_{\mathrm{nm}}^2 \mathrm{dx}$$

Donde δ_{ij} es la delta de Kronecker de índices i,j. Desarrollando la integral

$$\int_a^b y_{\mathrm{nm}} y_{\mathrm{pq}} \mathrm{dx} = \delta_{(n-m)(p-q)} \left(\frac{A}{2\lambda_{\mathrm{nm}}}\right) \left(\left(\frac{\sin\left(2\left(\lambda_{\mathrm{nm}} b + \phi\right)\right) - \sin\left(2\left(\lambda_{\mathrm{nm}} a + \phi\right)\right)}{2} \right) + \lambda_{\mathrm{nm}} \left(b - a\right) \right)$$

Donde λ_{nm} corresponde a los valores propios según cada condición que se muestra en a),b),c) y d).

ii) Gráfica de las Función correspondientes al primer, quinto, vigésimo y quincuagésimo valor propio, para a=0 y b=1.

Fijamos "n":

```
n = -1;
```

Introducimos los demás datos:

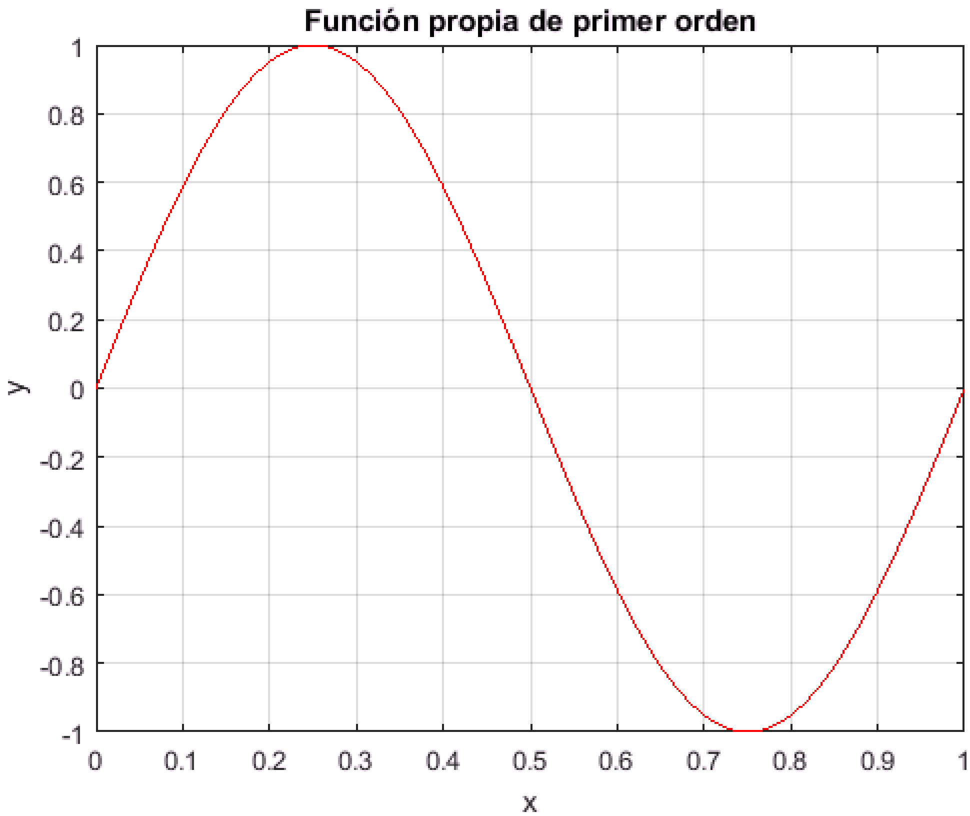
```
a = 0;
b = 1;
x = linspace(a,b,1000);
```

Imponemos una amplitud de A=1:

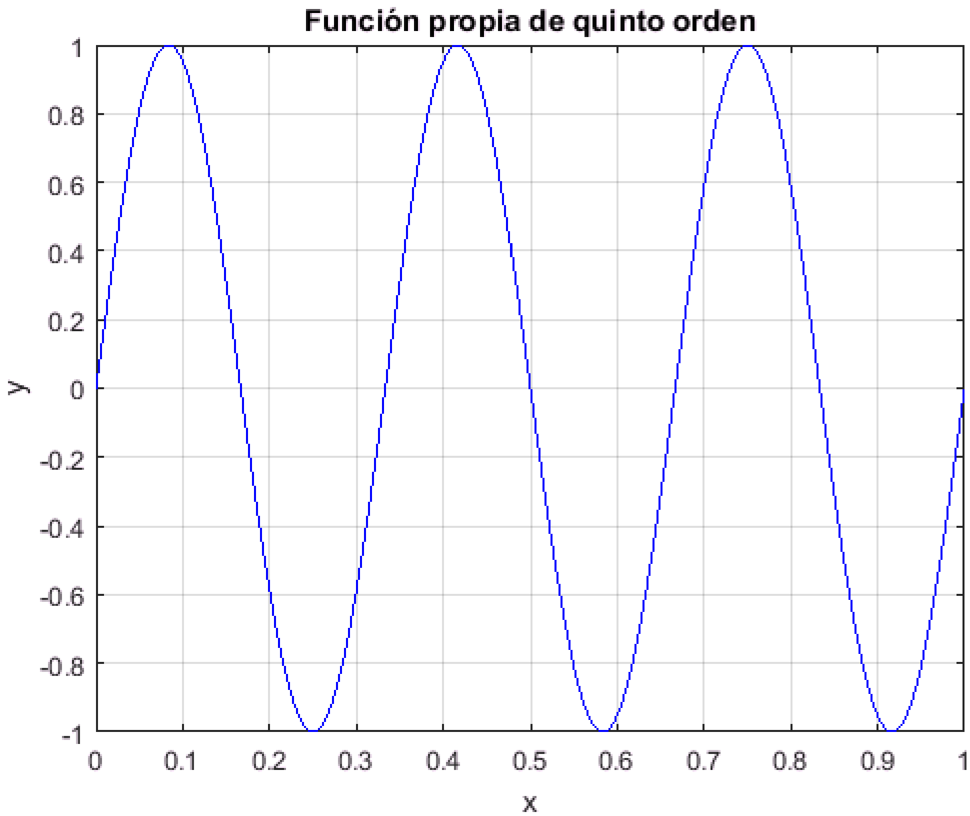
```
A = 1;
```

Para la condición a):

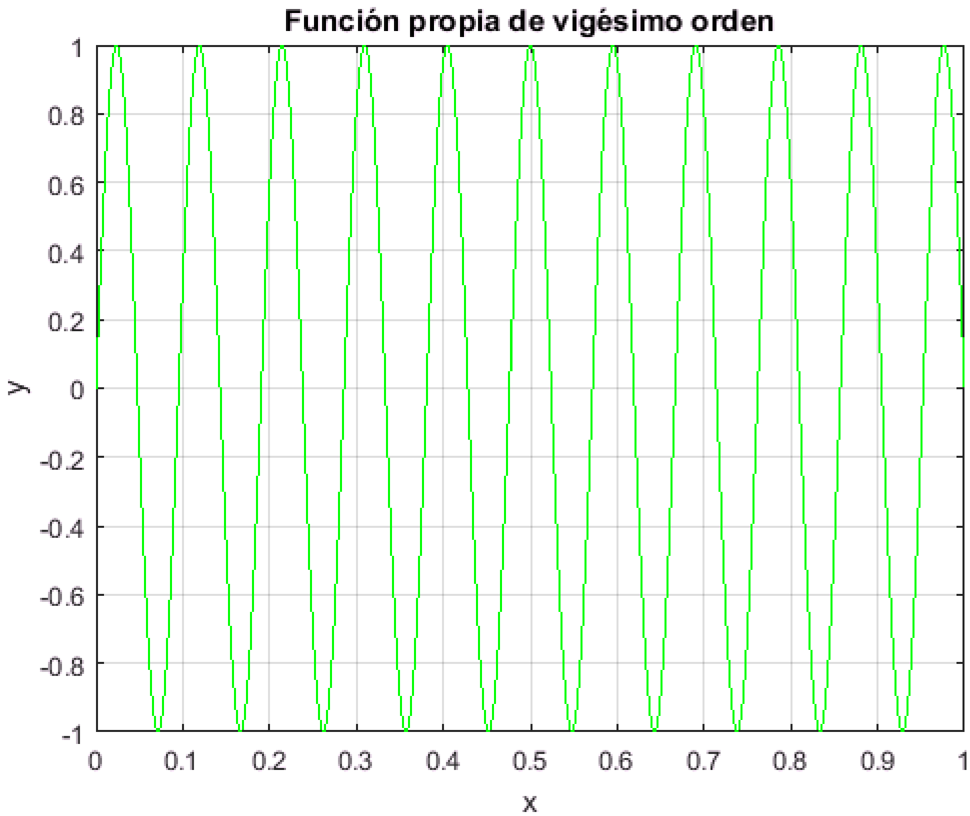
```
phi = ((2*n+1)/2)*pi;
lambda = @(m) ((((2*m+1)/2)*pi)-phi)/b;
y = @(x,m) A*cos(x*lambda(m)+phi);
plot(x,y(x,1),'r');
grid on
xlabel('x')
ylabel('y')
title('Función propia de primer orden')
```



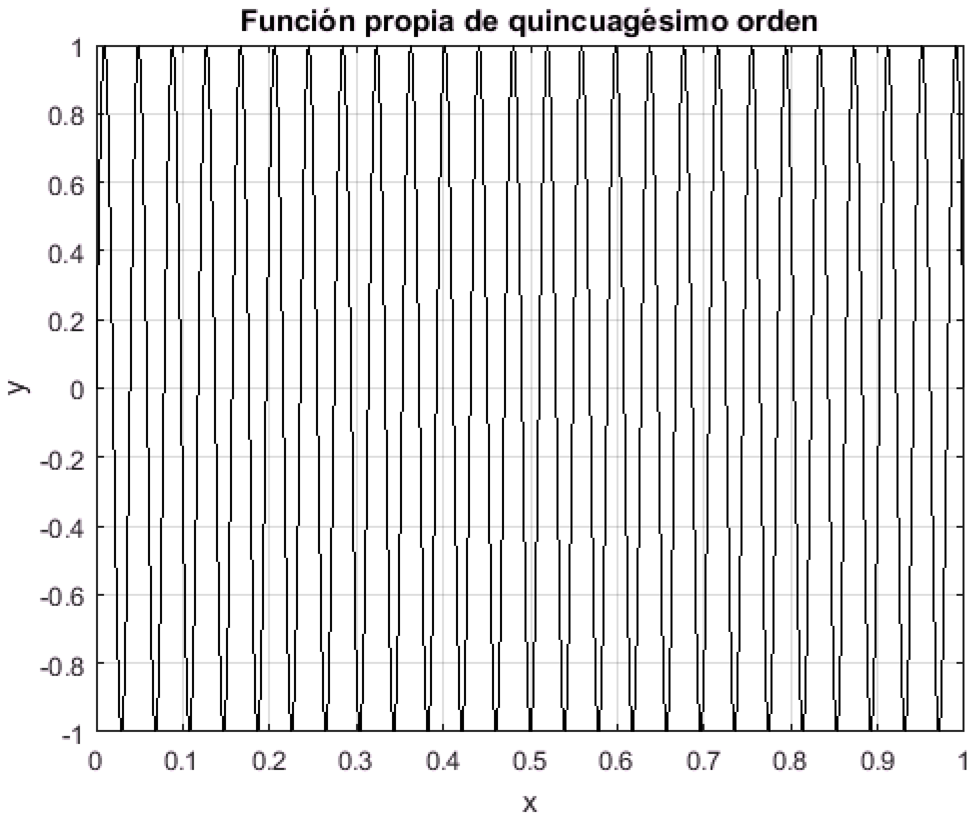

```
plot(x,y(x,5),'b');
grid on
xlabel('x')
ylabel('y')
title('Función propia de quinto orden')
```




```
plot(x,y(x,20),'g');
grid on
xlabel('x')
ylabel('y')
title('Función propia de vigésimo orden')
```

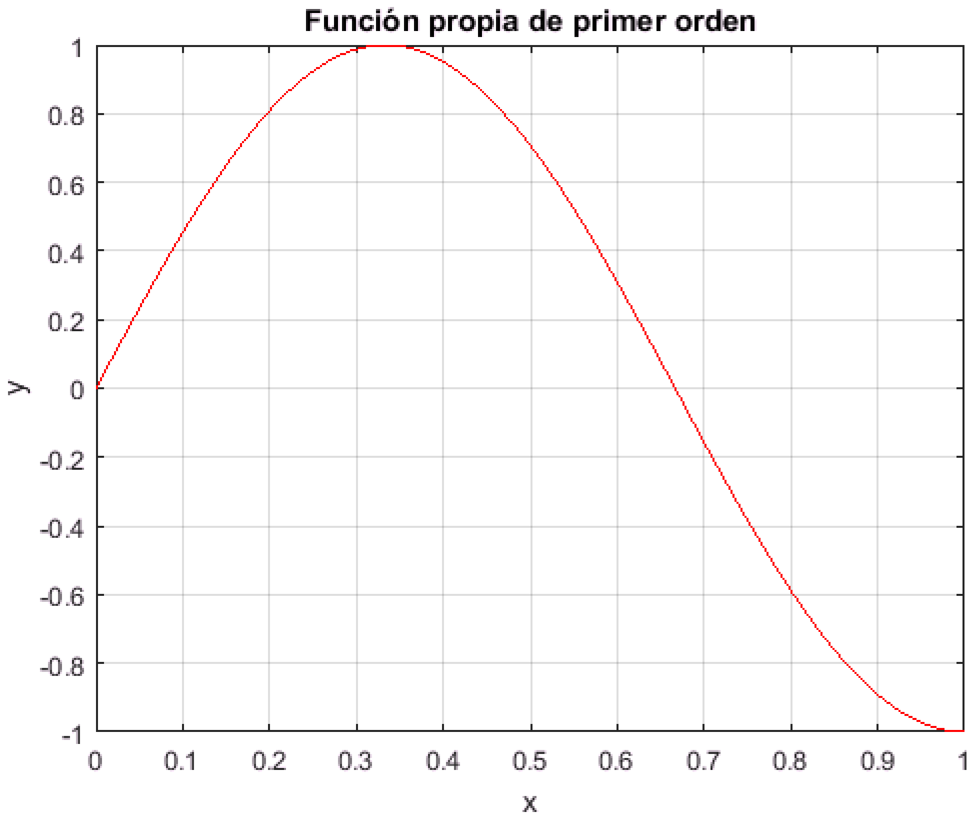



```
plot(x,y(x,50),'k');
grid on
xlabel('x')
ylabel('y')
title('Función propia de quincuagésimo orden')
```

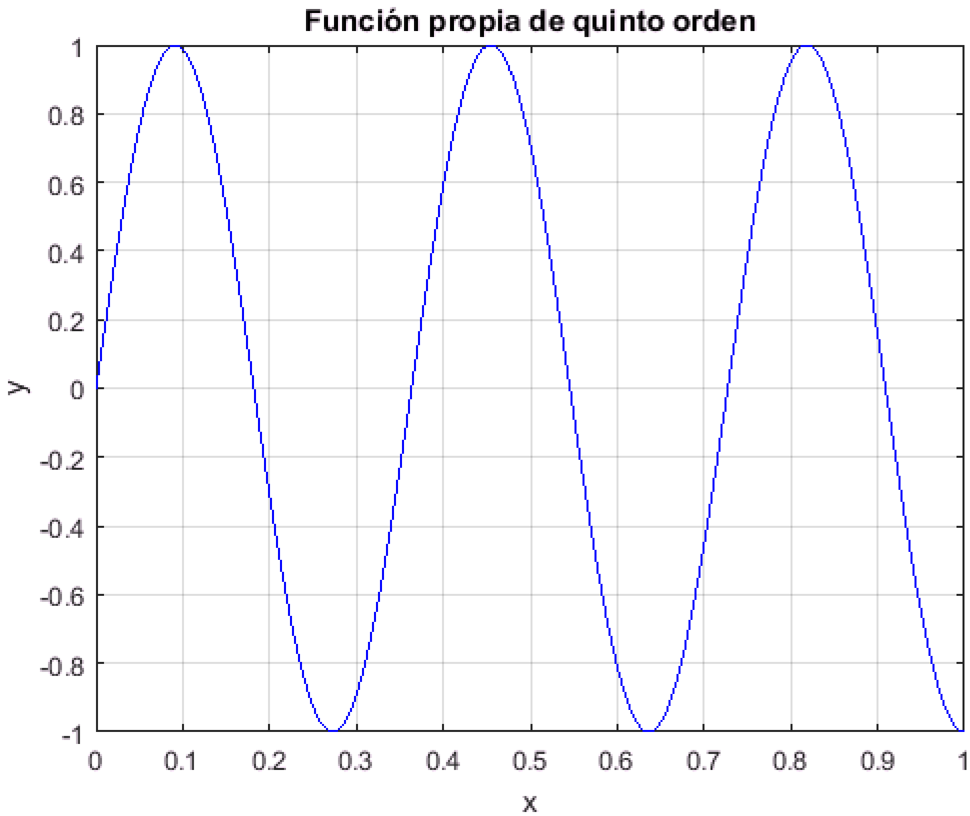


Para la condición b):

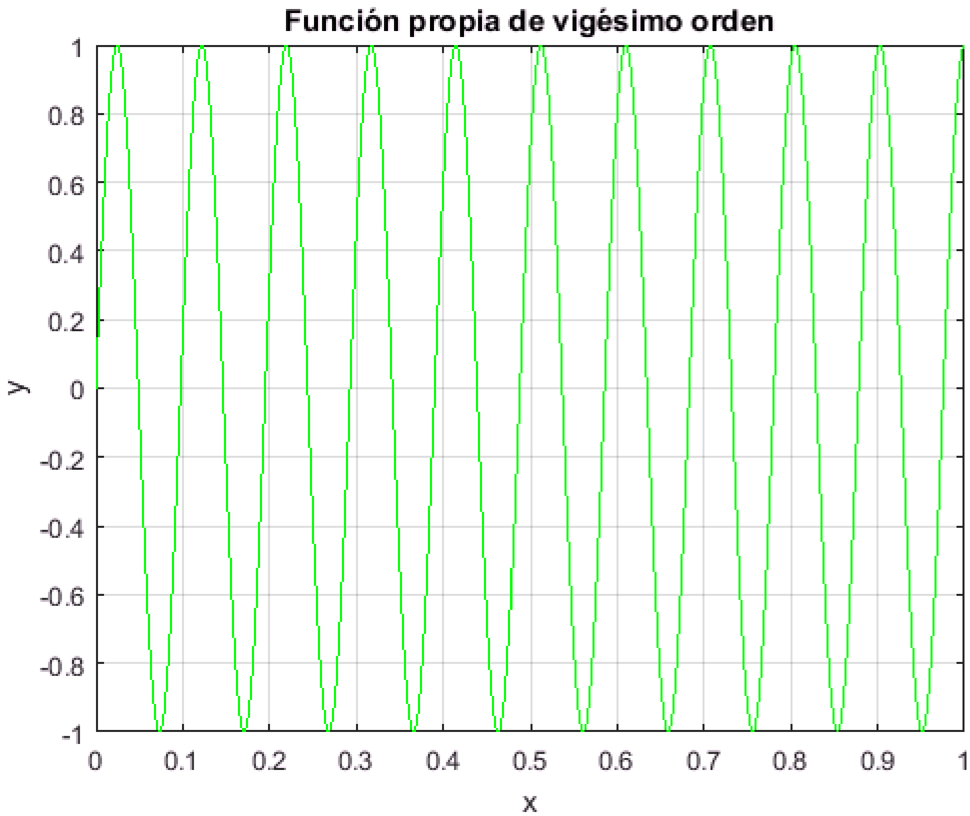
```
phi = ((2*n+1)/2)*pi;
lambda = @(m) ((m*pi)-phi)/b;
y = @(x,m) A*cos(x*lambda(m)+phi);
plot(x,y(x,1),'r');
grid on
xlabel('x')
ylabel('y')
title('Función propia de primer orden')
```



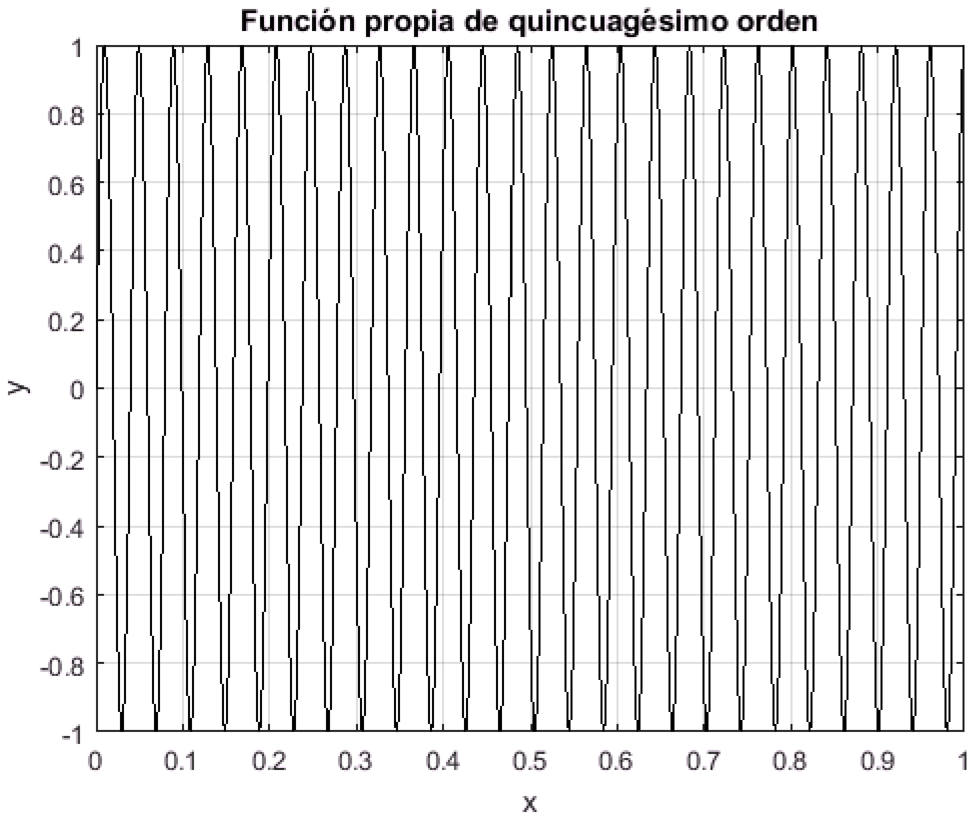

```
plot(x,y(x,5),'b');
grid on
xlabel('x')
ylabel('y')
title('Función propia de quinto orden')
```




```
plot(x,y(x,20),'g');
grid on
xlabel('x')
ylabel('y')
title('Función propia de vigésimo orden')
```




```
plot(x,y(x,50),'k');
grid on
xlabel('x')
ylabel('y')
title('Función propia de quincuagésimo orden')
```

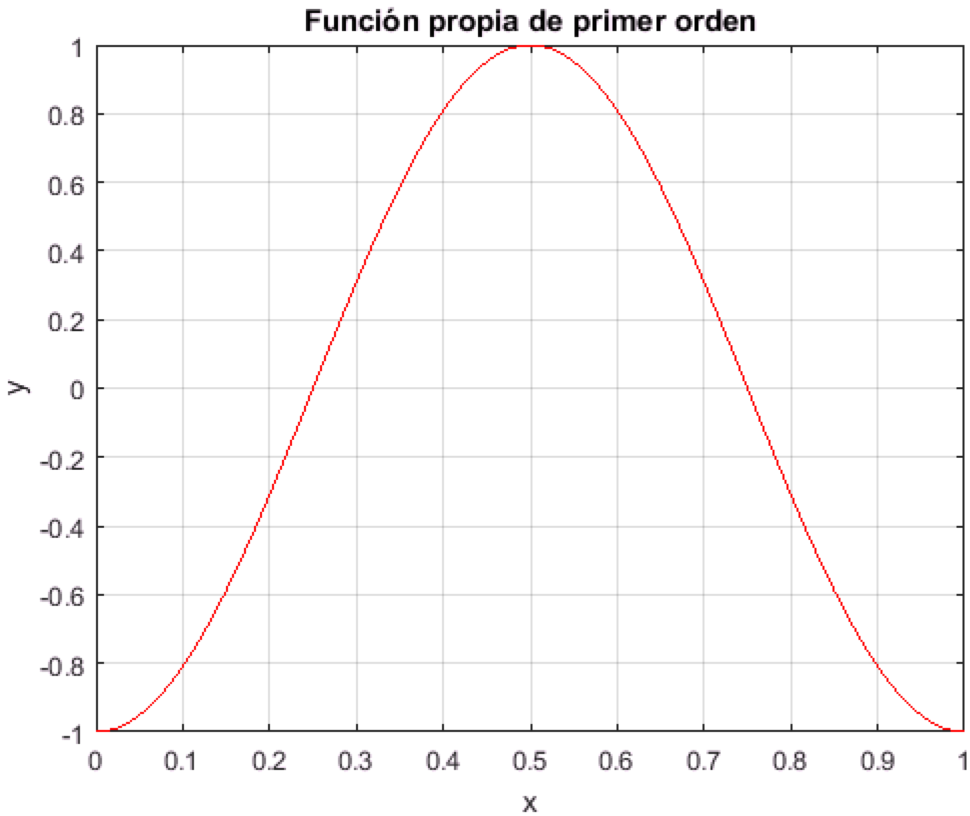


Para la condición c):

```
phi = n*pi
```

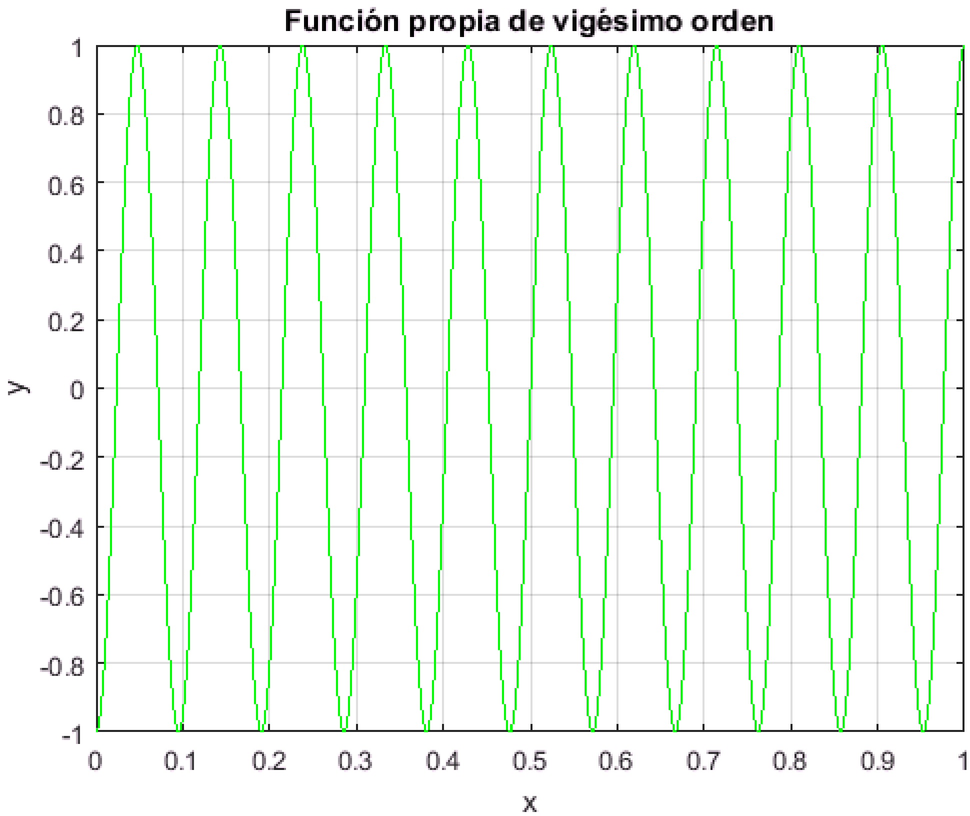
```
phi = -3.1416
```

```
lambda = @(m) ((m*pi)-phi)/b;
y = @(x,m) A*cos(x*lambda(m)+phi);
plot(x,y(x,1),'r');
grid on
xlabel('x')
ylabel('y')
title('Función propia de primer orden')
```

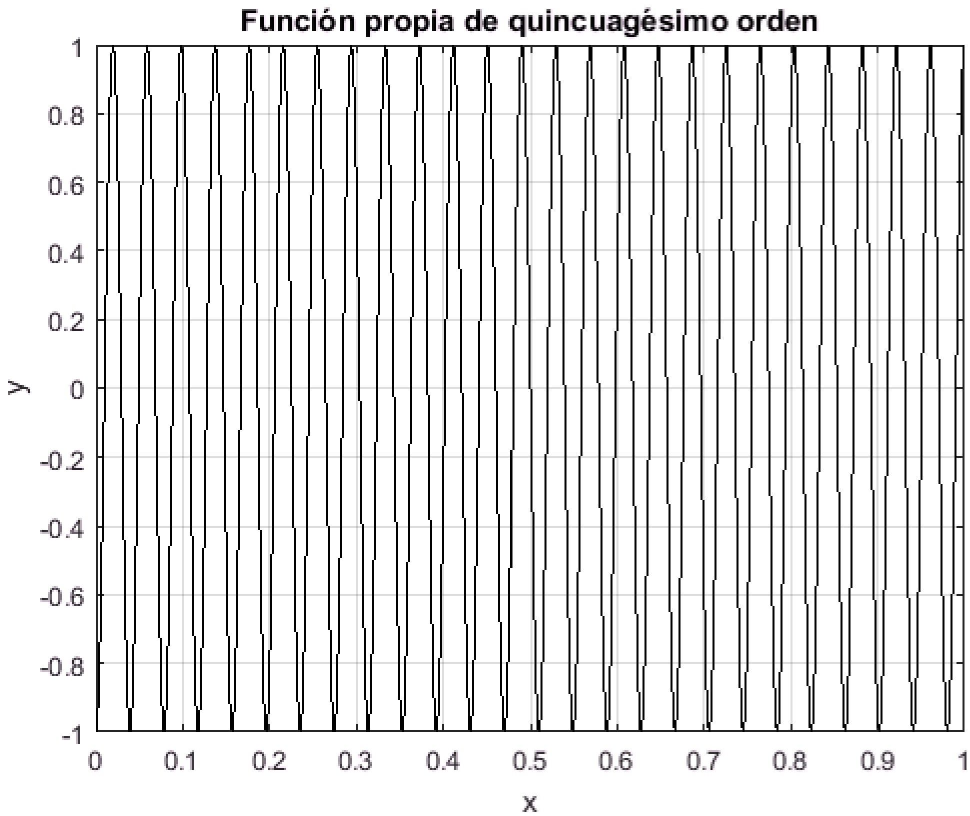



```
plot(x,y(x,5),'b');
grid on
xlabel('x')
ylabel('y')
title('Función propia de quinto orden')
```


```
plot(x,y(x,20),'g');
grid on
xlabel('x')
ylabel('y')
title('Función propia de vigésimo orden')
```




```
plot(x,y(x,50),'k');
grid on
xlabel('x')
ylabel('y')
title('Función propia de quincuagésimo orden')
```



<u>Para la condición d):</u>

Reemplazando para a = 0 y b = 1 en lo previamente obtenido:

$$\beta_1 \cos(\phi) + \lambda \sin(\phi) = 0$$
$$\beta_2 \cos(\lambda + \phi) + \lambda \sin(\lambda + \phi) = 0$$

Es claro que $\sin(\phi)$ debe ser diferente de cero, pues en caso contrario se llegaría a una contradicción. Luego:

$$\lambda = -\beta_1 \cot(\phi)$$

Reemplazando en la segunda igualdad:

$$\beta_2 \cos(\phi - \beta_1 \cot(\phi)) - \beta_1 \cot(\phi) \sin(\phi - \beta_1 \cot(\phi)) = 0$$

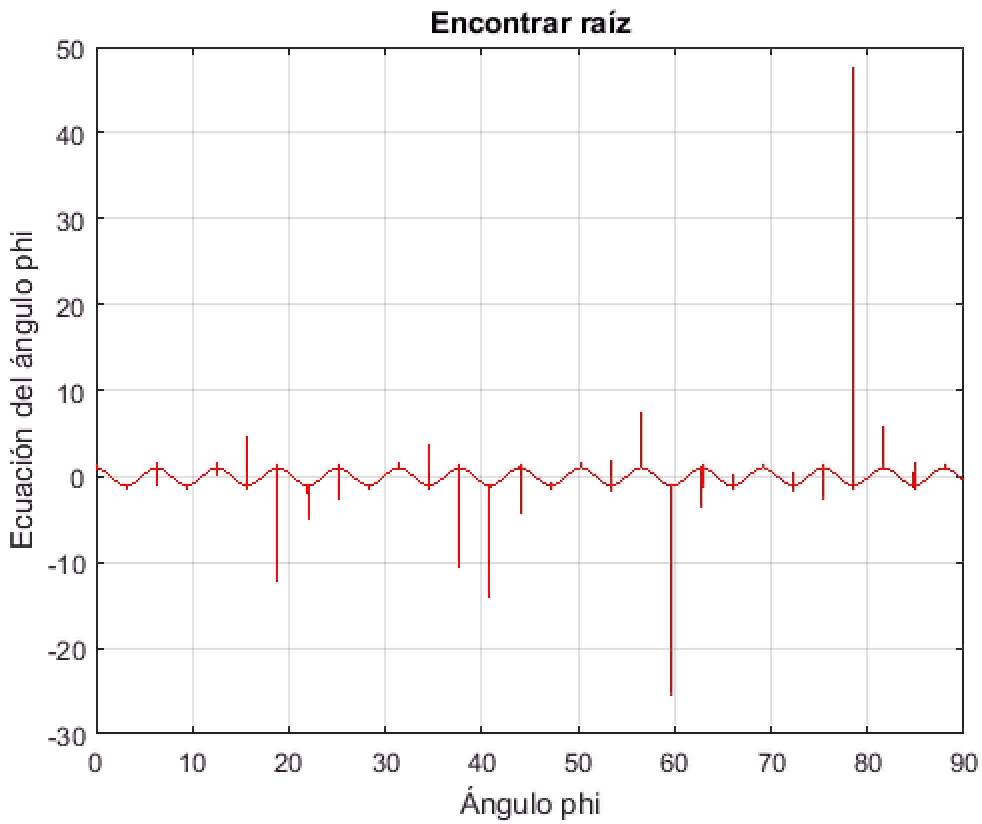
Esto es, $\lambda = \lambda_\phi$. El valor propio depende del ángulo ϕ. Con lo que debe resolverse la ecuación no lineal anterior usando métodos numéricos:

```
ecf = @(f,b1,b2) b2*cos(f-b1*cot(f))-b1*cot(f).*sin(f-b1*cot(f));
```

Introducimos los datos particulares:

```
b1 = 0.01;
b2 = 1;
inf = 0.01;
sup = 90;
f = [inf:0.01:sup];
```

```
ecfb1b2 = ecf(f,b1,b2);
plot(f,ecfb1b2,'r')
grid on
xlabel('Ángulo phi')
ylabel('Ecuación del ángulo phi')
title('Encontrar raíz')
```



Dado que es confuso determinar gráficamente una aproximación de la raíz, usamos la búsqueda incremental.

```
n = 0;
x0 = [];
for k = 1:length(f)-1
    if sign(ecfb1b2(k)) ~= sign(ecfb1b2(k+1))
        n = n + 1;
        x0(n,1) = f(k);
        x0(n,2) = f(k+1);
    end
end
```

Se encuentra (intervalo que encierra el cero):

```
x0
```

```
x0 =
    1.5700    1.5800
    4.7100    4.7200
    6.2700    6.2800
    6.2800    6.2900
    7.8500    7.8600
   10.9900   11.0000
   14.1300   14.1400
```

```
   15.7000   15.7100
   15.7100   15.7200
   17.2700   17.2800
      ⋮
```

Sujeto a los parámetros impuestos:

```
es = 0.00001;
maxit = 100;
```

Basta con tomar el ínfimo del intervalo hallado en cada fila, como nuestra primera aproximación de raíz. Se halla la cantidad de raíces encontradas según el número de intervalos hallados.

```
[f, c] = size(x0)
```

```
f = 57
c = 2
```

Sujetamos los parámetros:

```
maxiter = 100;
es = 0.0001;
necf = @(f) b2*cos(f-b1*cot(f))-b1*cot(f).*sin(f-b1*cot(f));
```

Hacemos uso del ***método numérico de la falsa posición modificada***:

```
for i = 1:f
    phi(i,1) = FPM(x0(i,1),x0(i,2),necf,maxiter,es);
end
```

Las raíces (los ángulos phi) vienen dadas, por tanto, por:

```
phi
```

```
phi =
    1.5708
    4.7124
    6.2796
    6.2816
    7.8540
   10.9956
   14.1372
   15.7044
   15.7115
   17.2788
      ⋮
```

```
lambda = @(f) -b1*cot(f);
```

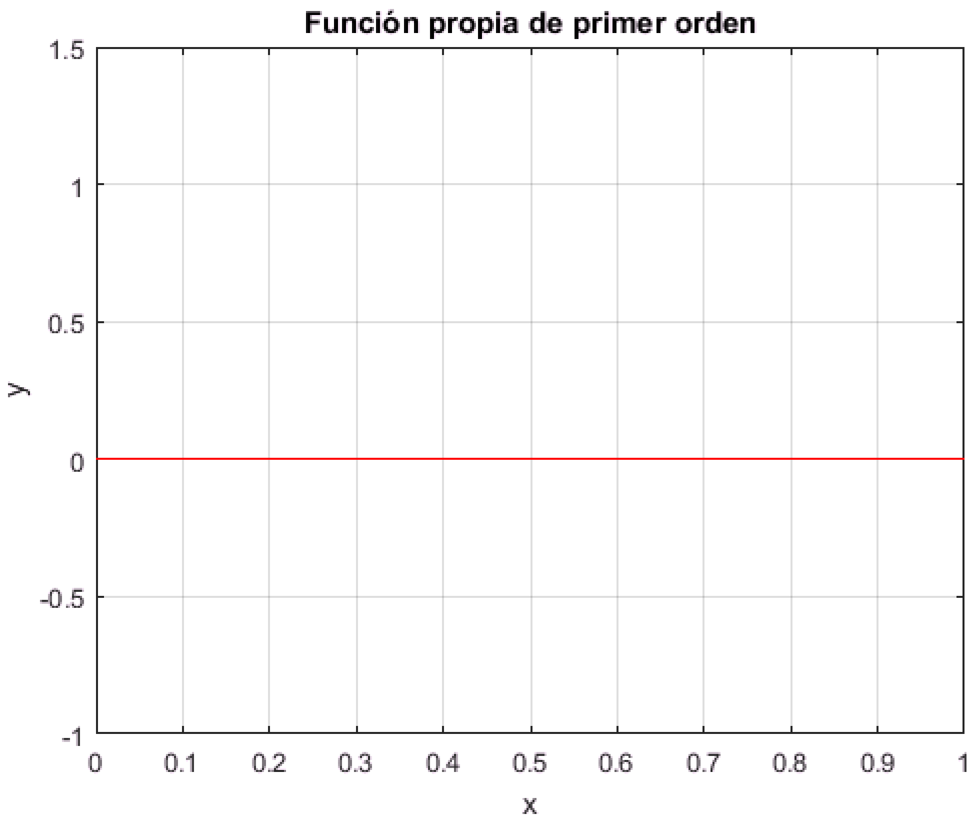
```
lambda = function_handle with value:
    @(f)-b1*cot(f)
```

El problema pide graficar para:

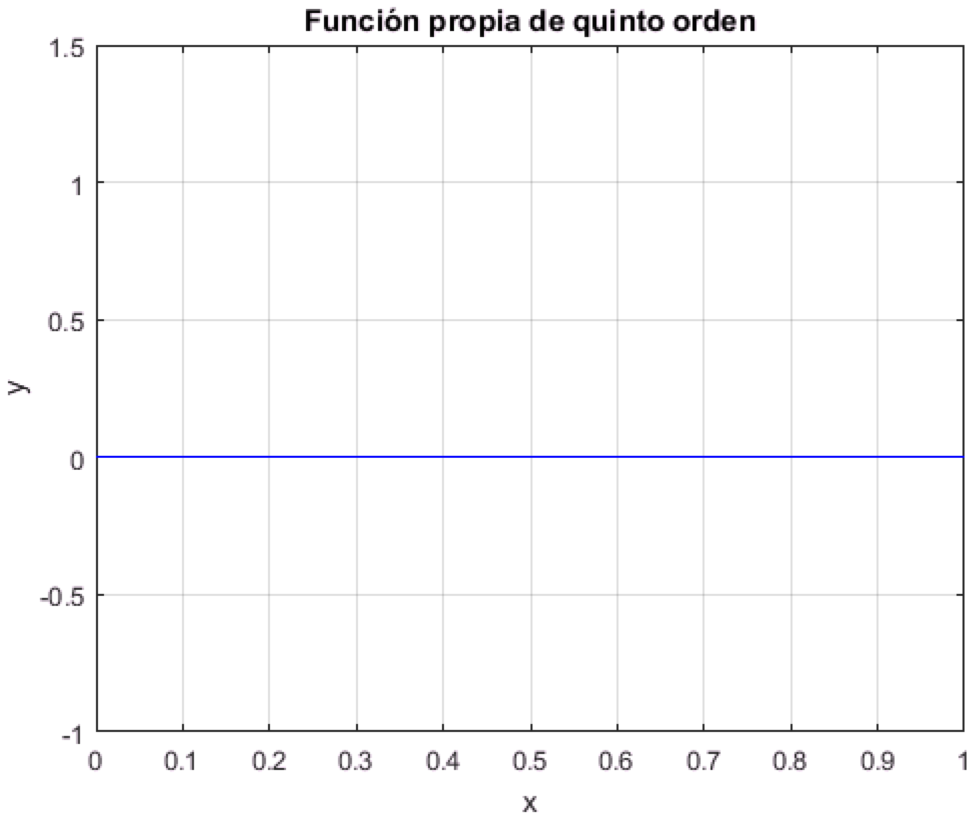
```
y = @(x,f) A*cos(x*lambda(f)+f);
```

Teniéndose:

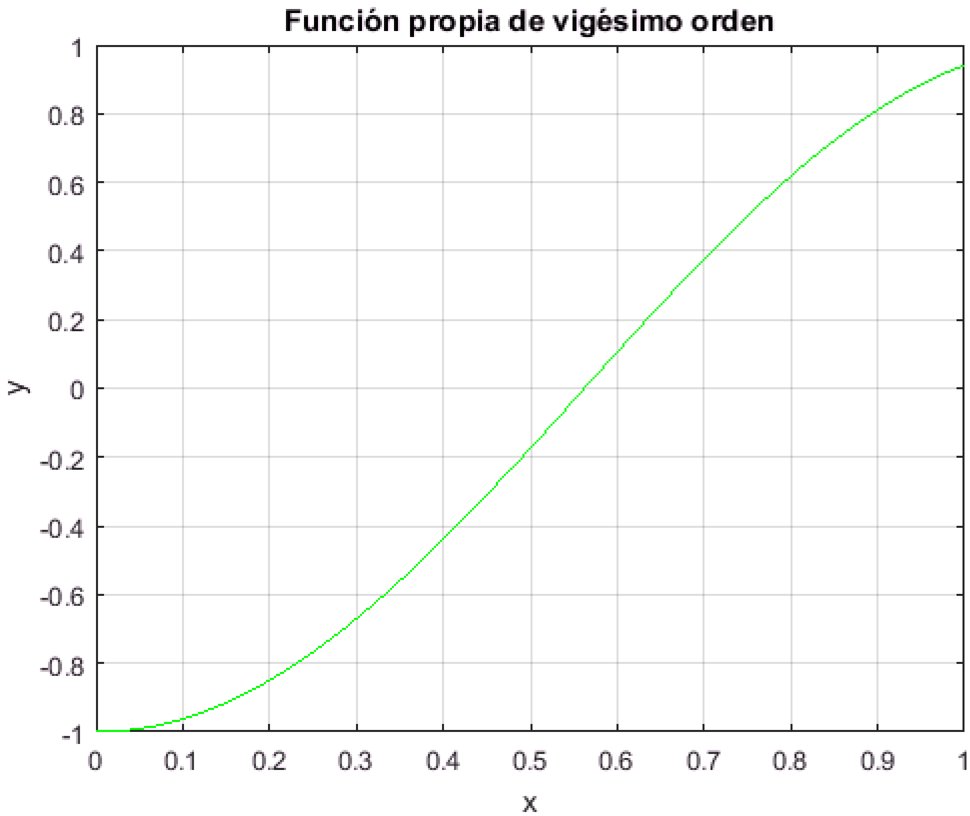
```
plot(x,y(x,phi(1,1)),'r');
grid on
xlabel('x')
ylabel('y')
title('Función propia de primer orden')
```



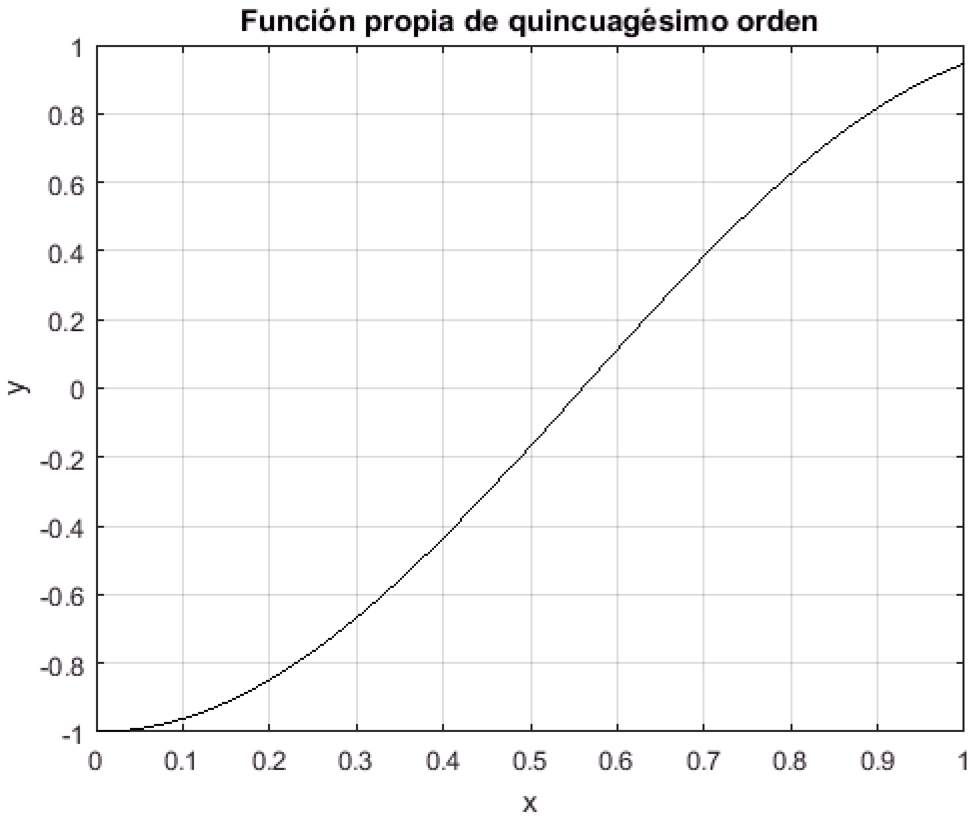

```
plot(x,y(x,phi(5,1)),'b');
grid on
xlabel('x')
ylabel('y')
title('Función propia de quinto orden')
```




```
plot(x,y(x,phi(20,1)),'g');
grid on
xlabel('x')
ylabel('y')
title('Función propia de vigésimo orden')
```




```
plot(x,y(x,phi(50,1)),'k');
grid on
xlabel('x')
ylabel('y')
title('Función propia de quincuagésimo orden')
```



PROBLEMA 2: EDP POR TRANSFORMADA DE LAPLACE

Problema 2

Use la transformada de Laplace para resolver el problema definido por la ecuación diferencial

$$\frac{\partial z}{\partial x} + \frac{\partial z}{\partial t} = z \qquad (1)$$

sujeta a las condiciones iniciales

$$z(x,0) = 1 \qquad (2)$$

$$z(0,t) = 1 \qquad (3)$$

y definida en $0 \le x < +\infty$, y $0 \le t < \infty$ (4)

Introducimos la notación:

$$L_t(z(x,t)) = Z(x,s)$$

Como la transformada de laplace respecto a la variable "t", para z=z(x,t). Recordamos también que:

$$L_t\left(\frac{\partial z}{\partial t}\right) = sZ(x,s) - z(x,0)$$

$$L_t\left(\frac{\partial z}{\partial x}\right) = \frac{d\,L_t\left(z(x,t)\right)}{\mathrm{dx}} = \frac{\mathrm{dZ}(x,s)}{\mathrm{dx}}$$

Luego, tomando transformada de laplace respecto a "t" a la EDP:

$$L_t\left(\frac{\partial z}{\partial x}\right) + L_t\left(\frac{\partial z}{\partial t}\right) = L_t(z)$$

$$\frac{\mathrm{dZ}(x,s)}{\mathrm{dx}} + \mathrm{sZ}(x,s) - z(x,0) = Z(x,s)$$

Usando la condición incial, y resumiendo la notación:

$$\frac{\mathrm{dZ}}{\mathrm{dx}} + \mathrm{sZ} - 1 = Z$$

Esto es:

$$\frac{\mathrm{dZ}}{\mathrm{dx}} + \mathrm{s}(Z - 1) = 1$$

$$\frac{\mathrm{dZ}}{\left(1 - \mathrm{s}(Z - 1)\right)} = \mathrm{dx}$$

Resolviendo:

$$\ln\left(\left(1 - \mathrm{s}(Z - 1)\right)\right) = -s\mathrm{x} + C$$

Evaluando para x = 0:

$$\ln\left(\left(1 - \mathrm{s}(Z(0,s) - 1)\right)\right) = C$$

Pero:

$$Z(0,s) = L_t\left(z(0,t)\right) = L_t(1) = \frac{1}{s}$$

Donde se usó la condición inicial, y lo conocido por tablas de transformadas de laplace. Esto es:

$$\ln\left(\left(1 - \mathrm{s}\left(\frac{1}{s} - 1\right)\right)\right) = \ln(s) = C$$

Es decir:

$$\ln\left(\left(1 - \mathrm{s}(Z - 1)\right)\right) = -s\mathrm{x} + \ln(s)$$

Se tiene, tomando exponenciales:

$$1 - \mathrm{s}(Z - 1) = se^{-s\mathrm{x}}$$

Es decir:

$$Z = L_t(z) = \frac{1}{s} - e^{-s\mathrm{x}} + 1$$

Y tomando transformada inversa:

$$z = L_t^{-1}\left(\frac{1}{s}\right) - L_t^{-1}\left(e^{-sx}\right) + L_t^{-1}(1)$$

Y usando tablas:

$$z = 1 - \delta(t - x) + \delta(t)$$

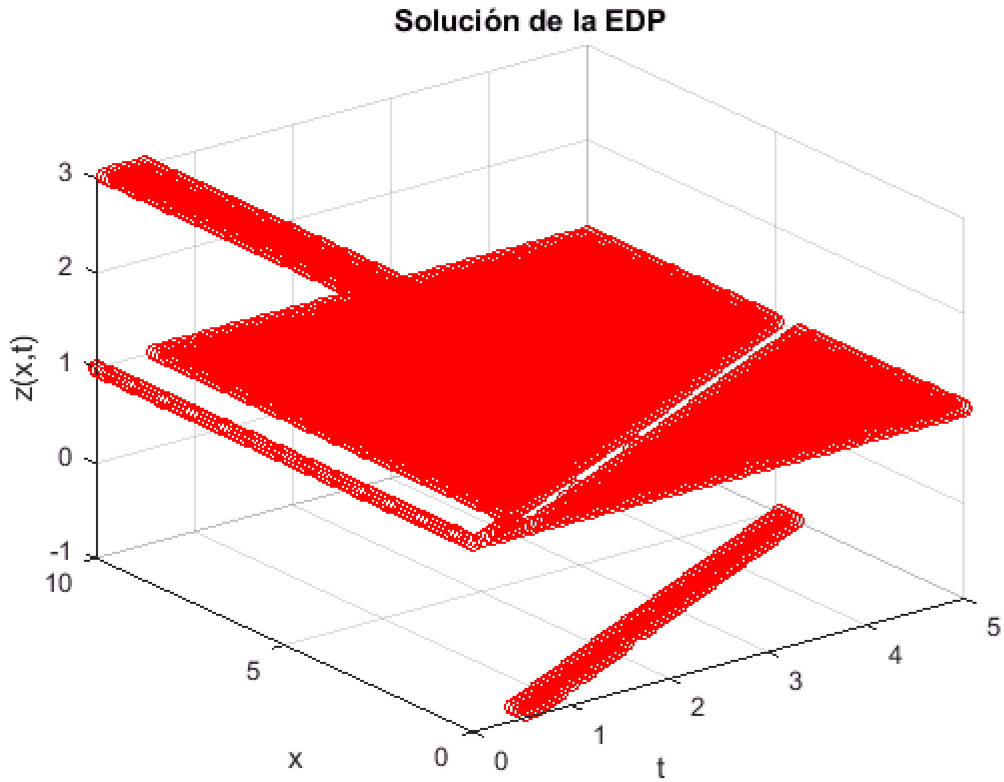
Donde $\delta(t)$ es la delta de Dirac.

GRÁFICA DE LA SOLUCIÓN:

```
tau = 0.5;
infi = 10;
delta = @(u) ((u>0)-(u>tau))/tau;
```

Introducimos las variables:

```
tsup = 5;
xsup = 10;
t = [0:0.1:tsup];
x = [0:0.1:xsup];
[X,T] = meshgrid(x,t);
z = 1-delta(T-X)+delta(T);
plot3(T,X,z,'ro');
shading interp
xlabel('t')
ylabel('x')
zlabel('z(x,t)')
title('Solución de la EDP')
grid on
```



MÉTODO DE LA FALSA POSICIÓN MODIFICADA

```
function [raiz] = FPM(xl, xu, F, maxiter, es)
    xr = xl;
    ea = 100;
    itera = 0;
    fl = F(xl);
    fu = F(xu);
```

Estos son los indicadores estacionarios:

```
    il = 0;
    iu = 0;
```

Continuamos con la ejecución del programa. Este es el proceso conocido de la falsa posición.

```
    for k=1:maxiter
        xrold = xr;
        xr = (xl*fu-xu*fl)/(fu-fl);
        fr = F(xr);
        itera = itera + 1;
        if xr ~= 0
            ea = abs((xr-xrold)/xr) * 100;
        end
```

Recordamos que "EncontrarRaiz" nos ayuda a suponer la existencia de una raíz (recordar Teorema de Bolzano).

```
        EncontrarRaiz = fl*fr;
```

Si es negativo, existe raíz.

```
        if EncontrarRaiz < 0
```

El cambio se hace en el límite superior, pues "EncontrarRaiz" se operó con f(xl) (el extremo inferior, que no variaría luego de

esta condicional; esto es, ESTÁ ESTACIONARIO).

```
            xu = xr;
            fu = F(xu);
```

El indicador que identifica el estacionamiento de "xu" se hace cero, pues ahora el extremo del intervalo ha cambiado.

Así, se le adiciona un valor más al contado que identifica el estacionamiento de "xl" por lo dicho anteriormente.

```
            iu = 0;
            il = il + 1;
```

Esta condicional verificará si es que "xl" permanece estacionario durante tres iteraciones consecutivas más.

```
            if il >= 2
```

Esta es la esencia del falsa posición modificada.

```
                fl= fl/2;
            end
```

El razonamiento es análogo para este caso.

```
        elseif EncontrarRaiz > 0
            xl = xr;
            fl = F(xl);
            il = 0;
            iu = iu + 1;
            if iu >= 2
                fu= fu/2;
            end
        else
            ea = 0;
        end
        if ea <= es
            break;
        end
    end
    raiz = xr;
end
```