

ALETA DE ENFRIAMIENTO

```
clear all
clearvars
clc
```

ECUACION DE VALORES PROPIOS

```
A = @(a,Nbi) a.*sin(a)-Nbi*cos(a)
```

```
A = function_handle with value:
    @(a,Nbi)a.*sin(a)-Nbi*cos(a)
```

```
dA = @(a,Nbi) a.*cos(a)+(Nbi+1)*sin(a)
```

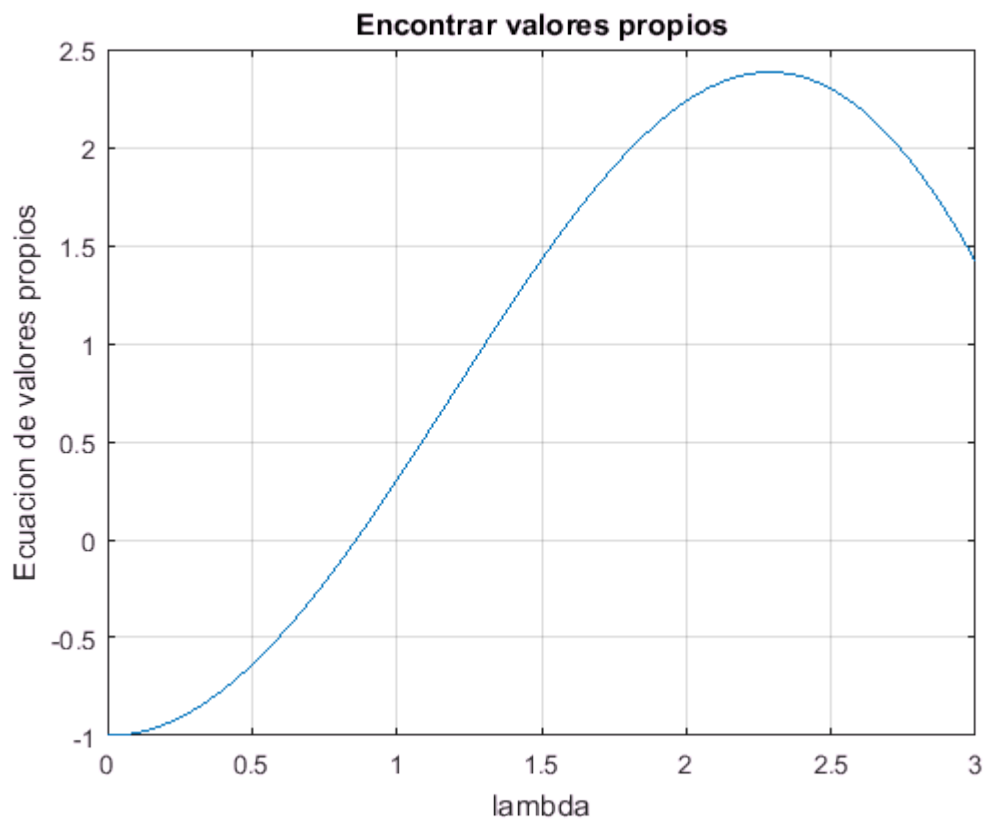
```
dA = function_handle with value:
    @(a,Nbi)a.*cos(a)+(Nbi+1)*sin(a)
```

MÉTODO GRÁFICO PARA LA DETERMINACIÓN DE VALORES PROPIOS

```
a = [0:0.01:3];
```

SE PROPONE UN NÚMERO DE BIOT:

```
Nbi = 1;
FA = A(a,Nbi);
plot(a,FA)
grid on
xlabel('lambda')
ylabel('Ecuacion de valores propios')
title('Encontrar valores propios')
```



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

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

Se encuentra (intervalo que encierra el cero):

x0

```
x0 =
    0.8600    0.8700
```

•

Sujeto a los parámetros impuestos:

```
es = 0.0000001;
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 = 1  
c = 2
```

Uso del método de Newton

```
for i = 1:f  
    roots(i,1) = NewtonRaphson(A,dA,x0(i,1),es,maxit,Nbi);  
end
```

Se encuentra la raíz

```
roots
```

```
roots = 0.8603
```

Se introduce el término n-ésimo de la serie hallada

```
uexact = @(y,l,r) ((4*sin(l).^2)./(l.*(2*l+sin(2*l)))).*(cosh(l*y)-tanh(l*r)*sinh(l*y))
```

```
uexact = function_handle with value:  
    @(y,l,r)((4*sin(l).^2)./(l.*(2*l+sin(2*l)))).*(cosh(l*y)-tanh(l*r)*sinh(l*y))
```

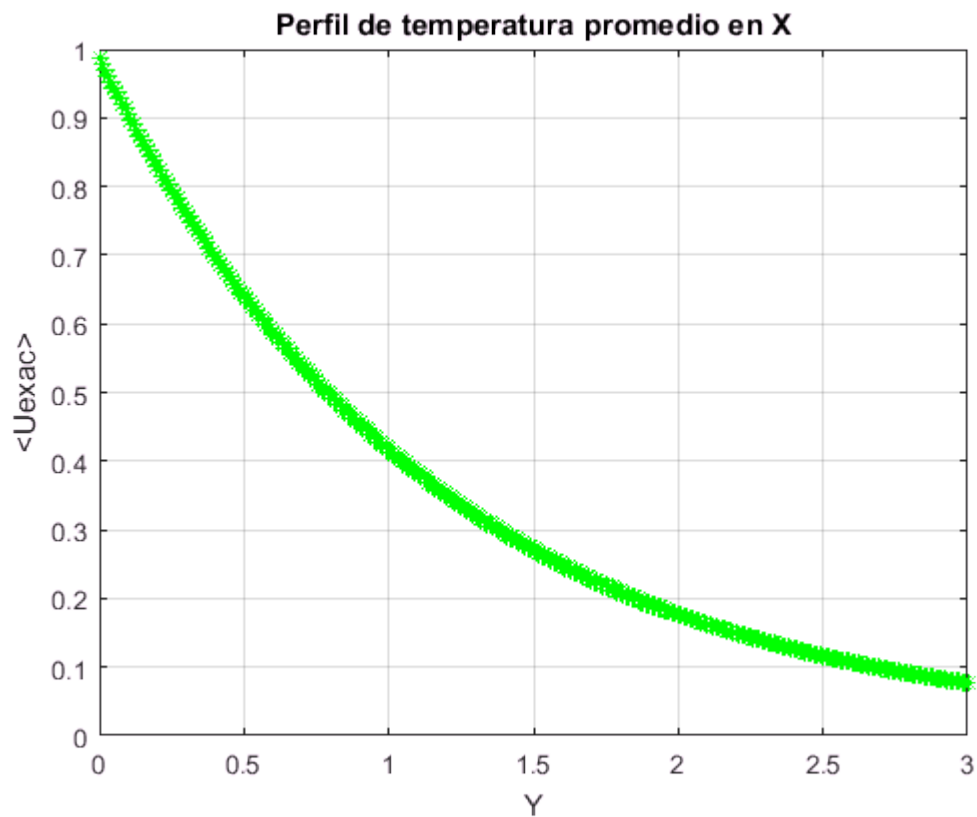
```
y = [0:0.01:3];
```

Se ejecuta la serie, para "f" términos ("f" es el número de raíces previamente halladas)

```
uex = 0;  
for i = 1:f  
    uex = uex + uexact(y,roots(f,1),5);  
end
```

OBSERVE QUE EL "5" EN EL ARGUMENTO DE LA uexact ANTERIOR ES $r = L/B$
(RELACIONADO CON UNA CONDICIÓN DE FRONTERA)

```
plot(y,uex,'g*')  
xlabel('Y')  
ylabel('<Uexac>')  
title('Perfil de temperatura promedio en X')  
grid on
```



ALGORITMO DEL MÉTODO DE NEWTON RAPHSON (Nbi es el numero de Biot a introducir)

```
function [root] = NewtonRaphson(f,df,x0,es,maxit,Nbi)
    iter = 0;
    xr = x0;
    while (1)
        xrold = xr;
        xr = xrold - f(xrold,Nbi)/df(xrold,Nbi);
        iter = iter + 1;
        if xr ~= 0
            ea = abs((xr-xrold)/xr)*100 ;
        end
        if ea <= es || iter >= maxit
            break;
        end
    end
    root = xr;
end
```