

# Universidad de Guanajuato

F.I.M.E.E.

## Laboratorio de Cálculo I

Prof. Ing. Daniel Arturo Razo Montes

### Práctica 5: Teorema de Rolle y comandos de control de flujo de programa

---

#### I. Introducción

En esta práctica se verá como comprobar las hipótesis del teorema de Rolle [ii) y iii)] usando el *toolbox* de matemática simbólica de MatLab.

#### II. Desarrollo

Teclee el siguiente listado en su editor de archivos *.m*. Las salidas del listado se verá en la ventana de comandos (*Command Window*).

```
clc           % Borra lo escrito y desplegado en la ventana de comandos
clear all    % Inicializa el espacio de trabajo en 0's; borra las variables
close all    % Cierra todas las ventanas abiertas (...de figuras)
             % previas
syms x t w   % declaracion de un objeto simbolico
v = [];      % variable auxiliar
% ----- Ejemplo 1 -----
y_1 = 4*x^3-9*x;           % funcion dada
disp('----- Ejemplo 1 -----');
disp('y_1(x) = ')
pretty(y_1)
disp('|')
disp('                Teorema de Rolle')
% En este problema la cota inferior es -1.5 y la superior es 1.5
a_1 = input('cota inferior: ') % captura una entrada en la ventana de
b_1 = input('cota superior: ') % comandos
disp('')
disp('Verificando las hipotesis del teorema de Rolle')
disp('i) derivada de la funcion y_1(x) = ')
der_1 = diff(y_1,x);       % derivada de una funcion
pretty(der_1)
y1 = sym2poly(y_1);        % pasa los coeficientes de una funcion
                             % polinomial a un vector
disp('ii) raices (cruces por el eje x) de la funcion y_1(x) = ')
r_1 = roots(y1)            % devuelve las raices de una funcion polinomial
v = [a_1 b_1];            % las cotas de entrada se guardan en un vector
% ciclo for para evaluar las 2 raices
for n = 1:2
    y_11(n) = 4*v(n).^3-9.*v(n); % funcion dada, igual que la simbolica,
end                          % se utilizara como auxiliar para evaluar
                             % la funcion en la cota inferior y superior

% condicional if para tomar la decision si se cumple o no la condicion iii)
if(y_11(1) ~= 0)
    disp('no se cumple la condicion iii, f(a) = f(b) = 0 !!!')
    break
else
    disp('se cumple la condicion iii), f(a) = f(b) = 0 !!!')
end

% encontrando el numero c que satisface la hipotesis del teorema
d1 = sym2poly(der_1);      % captura los coeficientes de la derivada
disp('raices que satistacen Dx(y(x)) = 0')
r_2 = roots(d1)
w = r_2;                   % variable auxiliar para evaluar la funcion
                             % en sus puntos criticos
```

```

for n = 1 : 2
    y_12(n) = 4*w(n).^3-9.*w(n); % funcion original evaluada en los puntos
                                % criticos
end
figure('Name','Ejemplo 1')
ezplot(y_1,[a_1 b_1])
hold on
grid
axis([-1.5 1.5 -8 8])
    for n = 1:2
        y_t(n) = y_12(n);           % variable auxiliar
        y_t1(n) = poly2sym(y_t(n)) % los elementos de un vector los pasa
                                % a una funcion polinomial simbolica
        ezplot(y_t1(n))
    end
title('y_{1} = 4*x^3-9*x')
disp('-----');
% -----

```

### Salida del listado 1:

```

----- Ejemplo 1 -----
y_1(x) =

```

$$4x^3 - 9x$$

|

Teorema de Rolle

cota inferior: -1.5

a\_1 =

-1.5000

cota superior: 1.5

b\_1 =

1.5000

Verificando las hipotesis del teorema de Rolle

i) derivada de la funcion y\_1(x) =

$$12x^2 - 9$$

ii) raices (cruces por el eje x) de la funcion y\_1(x) =

r\_1 =

0  
1.5000  
-1.5000

se cumple la condicion iii), f(a) = f(b) = 0 !!!

raices que satisfacen Dx(y(x)) = 0

r\_2 =

0.8660  
-0.8660

y\_t1 =

-3\*3^(1/2)

```
y_t1 =  
[ -3*3^(1/2), 3*3^(1/2) ]
```

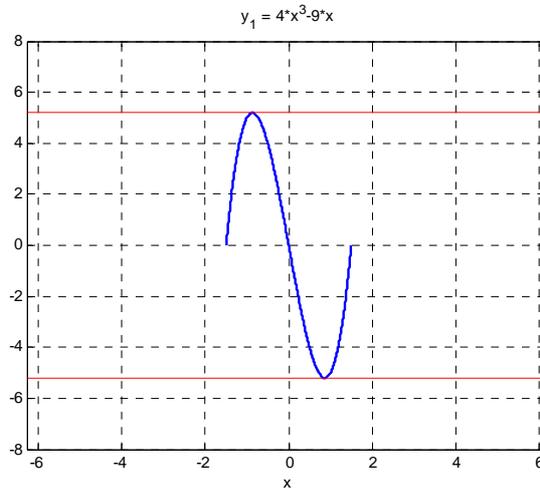


Figura 1. Gráfica de salida del listado 1

### III. Ejercicios

Desarrolle un programa similar al de esta práctica para probar el **teorema de valor medio de derivadas**, puede utilizar la función que sea apropiada para probar las hipótesis de éste teorema, es decir, puede elegir el problema 2, 3 o 4 de la sección 3.3 del libro Leithold, aunque no está restringido a ocupar solo este libro.

#### Ayuda

La función `input()` se utiliza para capturar el valor de una variable directamente del teclado (interacción con el usuario). La función `roots()` devuelve las raíces de una función polinomial, el comando `sym2poly()` regresa en un vector los coeficientes de una expresión polinomial simbólica mientras que el comando `poly2sym()` hace lo contrario, los elementos de un vector los hace coeficientes de una función polinomial simbólica. Puede ver como se utilizan estas funciones tecleando `help input`, `help roots`, `help sym2poly` y `help poly2sym`. Puede reciclar comandos de prácticas anteriores. En esta practica se vio la cláusula `if`, puede ver como se escribe tecleando `help if`. Esta palabra sirve para tomar decisiones de acuerdo a lo que se le ponga en su argumento, es decir, sirve para el control de flujo de programa.

### IV. Escriba sus conclusiones y observaciones

Nota: Reporte únicamente los ejercicios (sección III), poniendo el código fuente y los resultados de salida para cada uno (graficas y/o expresiones algebraicas). Escriba sus conclusiones generales sobre la práctica.