

**UNIVERSIDAD TECNOLÓGICA NACIONAL  
FACULTAD REGIONAL RESISTENCIA**



**INGENIERÍA EN SISTEMAS DE INFORMACIÓN**

**MATEMÁTICA SUPERIOR  
TRABAJO PRÁCTICO N° 2  
“Derivación Numérica”**

**Profesora:** García, Claudia Roxana

**Grupo Nro. 4 - Integrantes:**

- ACEVEDO, Fernando Enrique
- GETZEL, Martín Exequiel
- LUCAS, Dania

**2020**

## Parte A

```
function tabla = armarTabla (x0, h, f, n)
    if n<20
        error("El número n debe ser >= 20");
    else
        der=[];
        izq=[];
        if (mod(n,2))==0
            for i=1:((n/2)-1)
                izq(i)=x0-i*h;
            endfor
            for i=1:(n/2)
                der(i)=x0+i*h;
            endfor
        else
            for i=1:((n-1)/2)
                izq(i)=x0-i*h;
                der(i)=x0+i*h;
            endfor
        endif
        izq=sort(izq);
        X=[izq x0 der];
        Y=double(subs(f,X));
        tabla=[X ; Y];
    endif
endfunction;

#Referencias:
#fx0=f(x0)=f(x(0))
#fx0_1=f(x(0-1))
#fx0t1=f(x(0+1))
#fx0_2=f(x(0-2))
#fx0t2=f(x(0+2))
#f1x0=f'(x0)
#f2x0=f''(x0)

function mostrarValores(f1x0,Er1,Ep1,f2x0,Er2,Ep2)
    printf("-----\n");
    printf("Derivada primera= %d\n", double(f1x0));
    printf("Error relativo= %d\n", double(Er1));
    printf("Error porcentual= %d %%\n\n", double(Ep1));
    printf("-----\n");
    printf("Derivada segunda= %d\n", double(f2x0));
    printf("Error relativo= %d\n", double(Er2));
    printf("Error porcentual= %d %%\n", double(Ep2));
endfunction

function haciaADELANTE (x0, h, f, n, tabla)
    printf("\n\nDiferencias finitas hacia adelante\n");
    #armarTabla(x0, h, f, n)
    if (mod(n,2))==0
        #-----PAR
        #DATOS
        fx0t2=tabla(2,1:end)((n/2)+2);
        fx0t1=tabla(2,1:end)((n/2)+1);
        fx0=tabla(2,1:end)((n/2));
    else
        #-----IMPAR
        #DATOS
        fx0t2=tabla(2,1:end)(((n-1)/2)+3);
        fx0t1=tabla(2,1:end)(((n-1)/2)+2);
    end
end
```

```

        fx0=tabla(2,1:end)((n-1)/2)+1);
    endif
#-----Derivada primera
    f1x0=(fx0t1-fx0)/h;
#Error relativo y porcentual
    Er1=( subs(diff(f),x0)-f1x0 ) / subs(diff(f),x0);
    Ep1= Er1 * 100;
#-----Derivada segunda
    f2x0= ( fx0t2-2*fx0t1+fx0 )/(h^2);
#Error relativo y porcentual
    Er2=( subs(diff(diff(f)),x0)-f2x0 ) / subs(diff(diff(f)),x0);
    Ep2= Er2 * 100;
    mostrarValores(f1x0,Er1,Ep1,f2x0,Er2,Ep2);

endfunction

#-----

function haciaATRAS (x0, h, f, n, tabla)
    printf("\n\nDiferencias finitas hacia atras\n");
    #armarTabla(x0, h, f, n)
    if (mod(n,2))==0
        #-----PAR
        #DATOS
        fx0_2=tabla(2,1:end)((n/2)-2);
        fx0_1=tabla(2,1:end)((n/2)-1);
        fx0=tabla(2,1:end)((n/2));
    else
        #-----IMPAR
        #DATOS
        fx0_2=tabla(2,1:end)((n-1)/2-1);
        fx0_1=tabla(2,1:end)((n-1)/2);
        fx0=tabla(2,1:end)((n-1)/2+1);
    endif
#-----Derivada primera
    f1x0= ( fx0 - fx0_1 ) / h;
#Error relativo y porcentual
    Er1=( subs(diff(f),x0)-f1x0 ) / subs(diff(f),x0);
    Ep1= Er1 * 100;
#-----Derivada segunda
    f2x0= ( fx0 - 2*fx0_1 + fx0_2 )/(h^2);
#Error relativo y porcentual
    Er2=( subs(diff(diff(f)),x0)-f2x0 ) / subs(diff(diff(f)),x0);
    Ep2= Er2 * 100;
    mostrarValores(f1x0,Er1,Ep1,f2x0,Er2,Ep2);

endfunction

#-----

function CENTRALES (x0, h, f, n, tabla)
    #armarTabla(x0, h, f, n)
    printf("\n\nValores centrales\n");
    if (mod(n,2))==0
        #-----PAR
        #DATOS
        fx0_1=tabla(2,1:end)((n/2)-1);
        fx0t1=tabla(2,1:end)((n/2)+1);
        fx0=tabla(2,1:end)((n/2));
    else
        #-----IMPAR

```

```

        #DATOS
        fx0_1=tabla(2,1:end)((n-1)/2));
        fx0t1=tabla(2,1:end)((n-1)/2)+2);
        fx0=tabla(2,1:end)((n-1)/2)+1);
    endif
#-----Derivada primera
    f1x0= ( fx0t1 - fx0_1 ) / (2*h);
#Error relativo y porcentual
    Er1=( subs(diff(f),x0)-f1x0 ) / subs(diff(f),x0);
    Ep1= Er1 * 100;
#-----Derivada segunda
    f2x0= ( fx0t1 - 2*fx0 + fx0_1 )/(h^2);
#Error relativo y porcentual
    Er2=( subs(diff(diff(f)),x0)-f2x0 ) / subs(diff(diff(f)),x0);
    Ep2= Er2 * 100;
    mostrarValores(f1x0,Er1,Ep1,f2x0,Er2,Ep2);
endfunction

#-----
function derivacionNumerica (x0, h, f, n) #Con esta función se ejecutan el resto de métodos
    warning('off','all');
    tabla=armarTabla(x0, h, f, n);
    haciaADELANTE(x0, h, f, n, tabla);
    haciaATRAS(x0, h, f, n, tabla);
    CENTRALES(x0, h, f, n, tabla);
    warning('on','all');
endfunction

```

---

#### COMPROBACIÓN EN OCTAVE:

```

>> derivacionNumerica (5, 0.1, -x^2+4*x+1, 22)

Diferencias finitas hacia adelante
-----
Derivada primera= -6.1
Error relativo= -0.0166667
Error porcentual= -1.66667 %

-----
Derivada segunda= -2
Error relativo= 0
Error porcentual= 0 %

Diferencias finitas hacia atras
-----
Derivada primera= -5.9
Error relativo= 0.0166667
Error porcentual= 1.66667 %

-----
Derivada segunda= -2
Error relativo= 0
Error porcentual= 0 %

Valores centrales
-----
Derivada primera= -6
Error relativo= 0
Error porcentual= 0 %

-----
Derivada segunda= -2
Error relativo= 0
Error porcentual= 0 %

```

---

```
>> derivacionNumerica (7, 0.5, e^((x/4)+1), 24)
```

```
Diferencias finitas hacia adelante
```

```
-----  
Derivada primera= 4.16558  
Error relativo= -0.0651876  
Error porcentual= -6.51876 %
```

```
-----  
Derivada segunda= 1.10928  
Error relativo= -0.134625  
Error porcentual= -13.4625 %
```

```
Diferencias finitas hacia atras
```

```
-----  
Derivada primera= 3.67612  
Error relativo= 0.0599747  
Error porcentual= 5.99747 %
```

```
-----  
Derivada segunda= 0.86391  
Error relativo= 0.116354  
Error porcentual= 11.6354 %
```

```
Valores centrales
```

```
-----  
Derivada primera= 3.92085  
Error relativo= -0.00260607  
Error porcentual= -0.260607 %
```

```
-----  
Derivada segunda= 0.978938  
Error relativo= -0.00130286  
Error porcentual= -0.130286 %
```

---

```
>> derivacionNumerica (2.5, 0.3, cos(x/2), 25)
```

```
Diferencias finitas hacia adelante
```

```
-----  
Derivada primera= -0.484517  
Error relativo= -0.0211278  
Error porcentual= -2.11278 %
```

```
-----  
Derivada segunda= -0.0424122  
Error relativo= 0.461983  
Error porcentual= 46.1983 %
```

```
Diferencias finitas hacia atras
```

```
-----  
Derivada primera= -0.460913  
Error relativo= 0.0286196  
Error porcentual= 2.86196 %
```

```
-----  
Derivada segunda= -0.113187  
Error relativo= -0.43582  
Error porcentual= -43.582 %
```

```
Valores centrales
```

```
-----  
Derivada primera= -0.472715  
Error relativo= 0.00374578  
Error porcentual= 0.374578 %
```

```
-----  
Derivada segunda= -0.0786829  
Error relativo= 0.00187382  
Error porcentual= 0.187382 %
```

## Parte B

```
function integracion_Numerica(f,desde,hasta)
    syms x;
    warning('off','all');
    printf("\n\n-----Integraldefinida-----\n\n");
    pretty(f);
    printf("\n----- de x= %d a x= %d-----\n",double(desde),double(hasta));
    ##METODOS
    metTrapecios(f,desde,hasta,(hasta-desde)/3, 3);
    metTrapecios(f,desde,hasta,(hasta-desde)/9, 9);
    Simpson3(f,desde,hasta,(hasta-desde)/2, 2);
    Simpson3(f,desde,hasta,(hasta-desde)/10,10);
    Simpson8(f,desde,hasta,(hasta-desde)/3, 3);
    printf("-----\n");
endfunction

function Mostrar (titulo,tabla,AAprox,AExact>Error,n)
    printf("%s \n",titulo);
    Tabla=tabla
    printf("\nCon n= %d\n",double(n));
    printf("\n - Valor de Integral aproximada= %d \n", double(AAprox));
    printf("\n - Valor de Integral exacto= %d\n", double(AExact));
    printf("\n - Error porcentual= %d %%\n", double(Error));
endfunction

function Simpson8 (f,desde,hasta,h,n)
    silent_functions(1);
    warning('off','all');
    x=[];
    y=[];
    aux=desde;
    i=1;
    while (aux<=hasta)
        x(i)=aux;
        y(i)= double(subs(f,aux));
        aux=aux+h;
        i=i+1;
    endwhile
    tabla=[x ; y];
    sum=y(1)+y(columns(y));
    for i=2:columns(y)-1
        sum=sum+(3*y(i));
    endfor;
    syms x;
    AExact = int(f,x,desde,hasta);
    AAprox=(3/8)*h*sum;
    Ea=abs(AExact-AAprox); #Error absoluto
    Error=(Ea/AExact)*100;

    titulo="-----METODO DE SIMPSON
3/8-----\n\n";

    silent_functions(0);
    Mostrar(titulo,tabla,AAprox,AExact>Error,n);
endfunction
```

```

Function metTrapecios (f, desde, hasta, h, n)
    silent_functions(1);
    warning('off','all');
    x=[];
    y=[];
    aux=desde;
    i=1;
    while (aux<=hasta)
        x(i)=aux;
        y(i)= double(subs(f,aux));
        aux=aux+h;
        i=i+1;
    endwhile
    tabla=[x ; y];
    y0=y(1);
    sum=0;
    yn=y(columns(y));
    for i=2:columns(y)-1
        sum=sum+y(i)
    endfor;
    syms x;
    AAprox = 1/2*h*(y0+yn+2*sum);
    AExact = int(f,x,desde,hasta);
    Error=((desde-hasta)/AExact)*h^2*subs((diff(diff(f,x))),x,(hasta-desde)/2);
    titulo="-----METODO
TRAPECIOS-----\n\n";
    silent_functions(0);
    Mostrar(titulo,tabla,AAprox,AExact,Error,n);
endfunction

```

DE

---

**COMPROBACIÓN EN OCTAVE:**

```

-----Integral definida-----
      2
- x  + 4*x + 1
----- de x= 0 a x= 3-----
-----METODO DE TRAPECIOS-----

Tabla =

  0  1  2  3
  1  4  5  4

Con n= 3

- Valor de Integral aproximada= 11.5
- Valor de Integral exacto= 12
- Error porcentual= 0.5 %
-----METODO DE TRAPECIOS-----

Tabla =

Columns 1 through 7:

  0.00000  0.33333  0.66667  1.00000  1.33333  1.66667  2.00000
  1.00000  2.22222  3.22222  4.00000  4.55556  4.88889  5.00000

Columns 8 through 10:

  2.33333  2.66667  3.00000
  4.88889  4.55556  4.00000

- Valor de Integral aproximada= 11.9444
- Valor de Integral exacto= 12
- Error porcentual= 0.0555556 %
-----METODO DE SIMPSON 1/3-----

Tabla =

  0.00000  1.50000  3.00000
  1.00000  4.75000  4.00000

Con n= 2

- Valor de Integral aproximada= 12
- Valor de Integral exacto= 12
- Error porcentual= 0 %
-----METODO DE SIMPSON 1/3-----

Tabla =

Columns 1 through 7:

  0.00000  0.30000  0.60000  0.90000  1.20000  1.50000  1.80000
  1.00000  2.11000  3.04000  3.79000  4.36000  4.75000  4.96000

Columns 8 through 11:

  2.10000  2.40000  2.70000  3.00000
  4.99000  4.84000  4.51000  4.00000

Con n= 10

- Valor de Integral aproximada= 12
- Valor de Integral exacto= 12
- Error porcentual= 0 %
-----METODO DE SIMPSON 3/8-----

Tabla =

  0  1  2  3
  1  4  5  4

Con n= 3

- Valor de Integral aproximada= 12
- Valor de Integral exacto= 12
- Error porcentual= 0 %
-----

```



```

-----Integral definida-----
      x
      - + 1
      4
      e

----- de x= 1 a x= 2-----
-----METODO DE TRAPECIOS-----

Tabla =

      1.0000   1.3333   1.6667   2.0000
      3.4903   3.7937   4.1234   4.4817

Con n= 3

- Valor de Integral aproximada= 3.96768
- Valor de Integral exacto= 3.96538
- Error porcentual= -0.00539428 %
-----METODO DE TRAPECIOS-----

Tabla =

Columns 1 through 8:

      1.0000   1.1111   1.2222   1.3333   1.4444   1.5556   1.6667   1.7778
      3.4903   3.5887   3.6897   3.7937   3.9005   4.0104   4.1234   4.2395

Column 9:

      1.8889
      4.3589

Con n= 9

- Valor de Integral aproximada= 3.47449
- Valor de Integral exacto= 3.96538
- Error porcentual= -0.000599364 %
-----METODO DE SIMPSON 1/3-----

Tabla =

      1.0000   1.5000   2.0000
      3.4903   3.9551   4.4817

Con n= 2

- Valor de Integral aproximada= 3.96539
- Valor de Integral exacto= 3.96538
- Error porcentual= 0.000124025 %
-----METODO DE SIMPSON 1/3-----

Tabla =

Columns 1 through 8:

      1.0000   1.1000   1.2000   1.3000   1.4000   1.5000   1.6000   1.7000
      3.4903   3.5787   3.6693   3.7622   3.8574   3.9551   4.0552   4.1579

Columns 9 and 10:

      1.8000   1.9000
      4.2631   4.3710

Con n= 10

- Valor de Integral aproximada= 3.37889
- Valor de Integral exacto= 3.96538
- Error porcentual= 14.7903 %
-----METODO DE SIMPSON 3/8-----

Tabla =

      1.0000   1.3333   1.6667   2.0000
      3.4903   3.7937   4.1234   4.4817

Con n= 3

- Valor de Integral aproximada= 3.96539
- Valor de Integral exacto= 3.96538
- Error porcentual= 9.10133e-05 %
-----

```

```

-----Integral definida-----
      /x\
cos|-|
      \2/

----- de x= -1 a x= 3-----
-----METODO DE TRAPECIOS-----

Tabla =

-1.000000  0.333333  1.666667  3.000000
0.877583  0.986143  0.672412  0.070737

Con n= 3

- Valor de Integral aproximada= 2.84362
- Valor de Integral exacto= 2.95384
- Error porcentual= 0.325183 %
-----METODO DE TRAPECIOS-----

Tabla =

Columns 1 through 6:

-1.000000  -0.555556  -0.111111  0.333333  0.777778  1.222222
0.877583  0.961667  0.998457  0.986143  0.925331  0.819011

Columns 7 through 10:

1.666667  2.111111  2.555556  3.000000
0.672412  0.492744  0.288843  0.070737

Con n= 9

- Valor de Integral aproximada= 2.94168
- Valor de Integral exacto= 2.95384
- Error porcentual= 0.0361314 %
-----METODO DE SIMPSON 1/3-----

Tabla =

-1.000000  1.000000  3.000000
0.877583  0.877583  0.070737

Con n= 2

- Valor de Integral aproximada= 2.97243
- Valor de Integral exacto= 2.95384
- Error porcentual= 0.629429 %
-----METODO DE SIMPSON 1/3-----

Tabla =

Columns 1 through 6:

-1.000000  -0.600000  -0.200000  0.200000  0.600000  1.000000
0.877583  0.955336  0.995004  0.995004  0.955336  0.877583

Columns 7 through 11:

1.400000  1.800000  2.200000  2.600000  3.000000
0.764842  0.621610  0.453596  0.267499  0.070737

Con n= 10

- Valor de Integral aproximada= 2.95387
- Valor de Integral exacto= 2.95384
- Error porcentual= 0.000879468 %
-----METODO DE SIMPSON 3/8-----

Tabla =

-1.000000  0.333333  1.666667  3.000000
0.877583  0.986143  0.672412  0.070737

Con n= 3

- Valor de Integral aproximada= 2.96199
- Valor de Integral exacto= 2.95384
- Error porcentual= 0.276016 %
-----

```

**Conclusión:** Teniendo en cuenta los resultados de todas las funciones calculadas, el método con mayor aproximación es Simpson  $\frac{1}{3}$ .