

Diagrama y Pseudocódigo. Primera y segunda derivada,  $f(x) = x^3 - 2x - 4$

Inicio

Declarar float fx(float x)

Declarar int i, float a=-2 b=2 h=0.25  
int n =  $(b-a)/h + 1$   
float x[n], fpa[n], fpb[n], fpc[n], fpa2[n], fpb2[n], fpc2[n]

For(i=0; i<n; i++)  
x[i] = a + i\*h

Primera derivada  
For(i=0; i<n; i++)  
fpa[i] =  $(f(x_{i+1}) - f(x_i)) / (x_{i+1} - x_i)$   
fpb[i] =  $(f(x_i) - f(x_{i-1})) / (x_i - x_{i-1})$   
fpc[i] =  $(f(x_{i+1}) - f(x_{i-1})) / (x_{i+1} - x_{i-1})$   
Imprimir resultados  
Repetir para Segunda derivada

Escribir la función  
float F  
F =  $x^3 - 2x - 4$   
return F

Fin

Importar bibliotecas stdio.h, math.h

float fx(float x)  
int i, n =  $(b-a)/h + 1$   
float x[n], fpa[n], fpb[n], fpc[n],  
fpa2[n], fpb2[n], fpc2[n]

For(i=0; i<n; i++)  
x[i] = a + i\*h

Primera derivada

For(i=0; i<n; i++)  
fpa[i] =  $(f(x_{i+1}) - f(x_i)) / h$   
fpb[i] =  $(f(x_i) - f(x_{i-1})) / h$   
fpc[i] =  $(f(x_{i+1}) - f(x_{i-1})) / 2h$

Segunda derivada (Repetir)

float fx(float x)  
float F  
F =  $(x * x * x) - (2 * x) - 4$   
return F

Fin