

Ejercicio

Contar las operaciones necesarias para evaluar el polinomio interpolante con el algoritmo realizado en la clase anterior.

```
function pz=interpolan(x,f,z)
c = [f(1)];
pz = f(1)*ones(size(z));
for j=2:k
    pj = evalnewton(c,x,x(j));
    cj = f(j)-pj;
    qz = ones(size(z));
    for i=1:j-1
        cj = cj/(x(j)-x(i));
        qz = qz.*(z-x(i));
    end
    c = [c cj];
    pz = pz + cj* qz;
end
endfunction
```

Ejercicio

Elaborar un programa que calcule la diferencia dividida de orden k de f en los puntos x_0, \dots, x_k usando la fórmula

$$C_k = \frac{f_k - P_{k-1}(x_k)}{Q_k(x_k)}$$

```
function c=diferencias_c(k,x,f)
c=[f(1)];
for j=1:k
    pj = evalnewton(c,x,x(j+1));
    cj = f(j+1)-pj;
    for i=1:j
        cj = cj/(x(j+1)-x(i));
    end
    c=[c cj];
end
endfunction
```

Ejercicio

Contar las operaciones necesarias para calcular cada una de las diferencias divididas del ejemplo anterior.

Ejercicio

Calcular las diferencias divididas hasta el orden 6 de la función $f(x) = x^2$ en los puntos $x_i = i + 1$ con $i = 0, \dots, 6$.

Ejercicio

Generalizar el resultado del ejercicio anterior.

Ejercicio

Contar las operaciones necesarias para calcular cada una de las diferencias divididas del ejemplo anterior.

Ejercicio

Calcular las diferencias divididas hasta el orden 6 de la función $f(x) = x^2$ en los puntos $x_i = i + 1$ con $i = 0, \dots, 6$.

Ejercicio

Generalizar el resultado del ejercicio anterior.

Ejercicio

Contar las operaciones necesarias para calcular cada una de las diferencias divididas del ejemplo anterior.

Ejercicio

Calcular las diferencias divididas hasta el orden 6 de la función $f(x) = x^2$ en los puntos $x_i = i + 1$ con $i = 0, \dots, 6$.

Ejercicio

Generalizar el resultado del ejercicio anterior.

Ejercicio

Las diferencias divididas cumplen una relación de recurrencia. Buscarla. Implementarla. Contar la cantidad de operaciones necesarias.

Ejercicio

Proponer un algoritmo para disminuir la cantidad de operaciones en `evalnewton.m`.

Ejercicio

Las diferencias divididas cumplen una relación de recurrencia. Buscarla. Implementarla. Contar la cantidad de operaciones necesarias.

Ejercicio

Proponer un algoritmo para disminuir la cantidad de operaciones en **evalnewton.m**.