

Primer Programa: Genera el polinomio interpolante

```
function p = PollInterNewton(x,y)
warning('off', 'all');
n = length(x);

#Construimos tabla de diferencias divididas
A = zeros(n, n+1);
A(:,1) = x(:);
A(:,2) = y(:);

for j = 1:n-1
    for i = j+1:n
        A(i,j+2) = (A(i,j+1) - A(i-1,j+1)) / (A(i,1) - A(i-j,1));
    endfor
endfor

coef = diag(A(:, 2:end)); #Usamos los coeficientes que están en la diagonal de la tabla de
diferencias divididas

#Construimos el polinomio simbólico
syms X;
p = coef(1);
term = 1;
for i = 2:n
    term = term * (X - x(i-1));
    p = p + coef(i) * term;
endfor

p = simplify(p); #Simplificamos
p = vpa(p, 6); #Devolverá el resultado en decimales (6 cifras)
endfunction
```

Segundo programa: Se le agrega los datos

```
pkg load symbolic

dias = [0, 6, 10, 13, 17, 20, 24, 28];
muestra1 = [6.67, 17.33, 42.67, 37.33, 30.10, 29.61, 29.31, 28.74];
muestra2 = [6.67, 16.11, 18.89, 15.00, 10.56, 9.44, 9.44, 8.89];

polmuestra1 = PollInterNewton(dias, muestra1);
polmuestra2 = PollInterNewton(dias, muestra2);

disp("Polinomio interpolador para muestra 1:");
pretty(polmuestra1)
fprintf("\n");
```

```
disp("Polinomio interpolador para muestra 2:");
pretty(polmuestra2)
fprintf("\n");
```