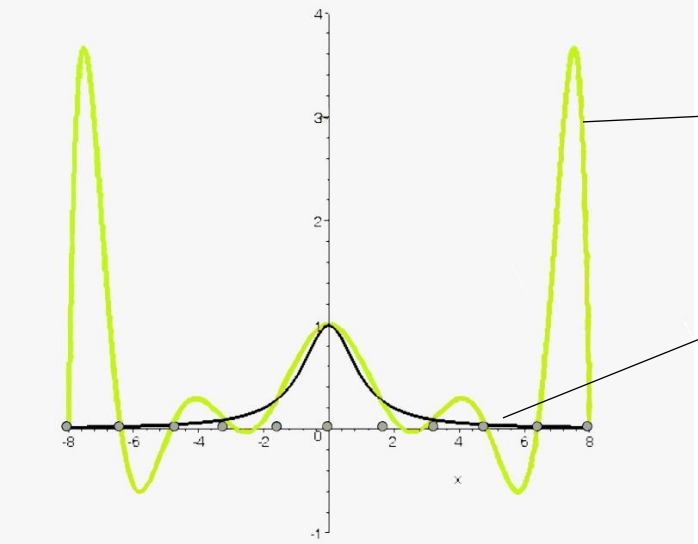
Interpolación a tramos

Tal y como hemos explicado en la página web, la interpolación por tramos se emplea cuando tenemos un polinomio de elevado grado, ya que puede no ser muy precisa la interpolación. Es por ello que dividimos el soporte en en sub-conjuntos de 2 o 3 puntos.

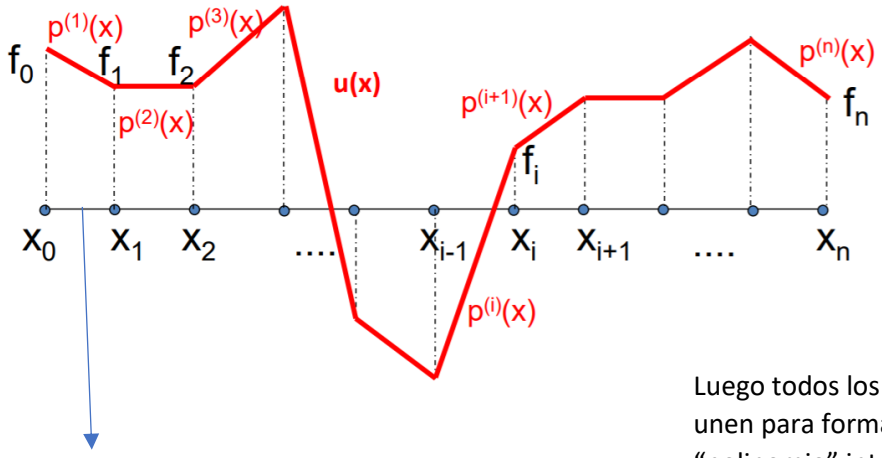


Como vemos, el polinomio al tener tanto grado, es muy variable.

Es por ello que el polinomio interpolador no es muy preciso

PERO TRANQUILO, no vas a aprender nada nuevo. Vas a emplear todos tus conocimientos sobre interpolación y los vas a juntar para hacer una interpolación, así que ¡vamos a ello!

En primer lugar deberíamos dividir el soporte en tramos de 2 o 3 puntos:



El primer tramo coge los puntos Xo y X1, por lo que interpolaremos en base a esos dos.

Luego todos los subtramos se unen para formar el "polinomio" interpolador. Esta función **NO ES POLINOMICA,** pero sí que lo son sus tramos.

De forma que si denominamos a nuestra función interpoladora u(x):

U(x)= { P1(x) si x ε [X0,X1]
P2(x) si x ε [X1,X2]
P3(X) si x ε [X2,X3]
Pi(x) si x ε [Xi-1,Xi]

El siguiente paso sería interpolarla mediante cualquiera de los métodos ya vistos.

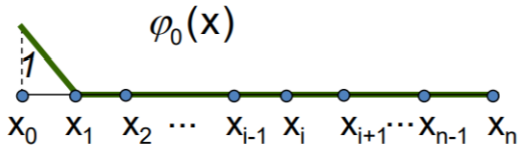
Es muy frecuente emplear el método de funciones de base:

u(x)=$\sum_{i=0}^{n} fi\varphi i(x)$=f0$\varphi$0(x)+f1$\varphi 1(x)$+f2$\varphi$2(x)…+fn$\varphi n(x)$

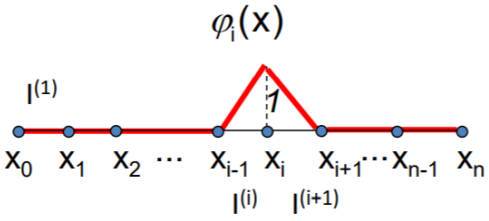
De forma que para cada subintervalo tenemos la siguiente fórmula general:

$$\varphi i(Xj) = \begin{cases} 1 \text{ si } i=j \\ 0 \text{ si } i\neq j \end{cases}$$

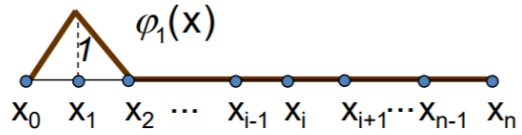
$$\varphi 0 = \begin{cases} \frac{X-X1}{X0-X1} \text{ si X } \varepsilon[X0,X1] \\ 0 \text{ si x } \varepsilon \text{ [X1,XN]} \end{cases}$$



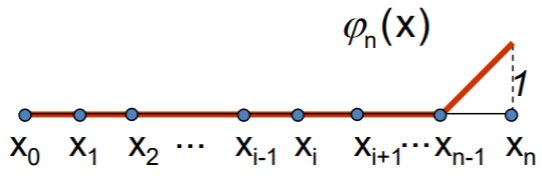
$$\varphi 1= \begin{cases} \frac{X-X0}{X1-X0} \text{ si X}\varepsilon \text{ [X0,X1]} \\ \frac{X-X2}{X1-X2} \; si \; X\varepsilon \text{ [X1,X2]} \\ 0 \text{ si X } \varepsilon \text{ [X2, Xn]} \end{cases}$$



$$\varphi_i= \begin{cases} \frac{x-xi-1}{(xi-xi-1)} & \text{si X}\varepsilon \text{ [Xi-1,Xi]} \\ \frac{x+xi+1}{xi+xi+1} & \text{Si X}\varepsilon \text{ [Xi, Xi+1]} \\ 0 & \text{si X}\varepsilon \text{ [}X_0, X_{i-1}\text{] U [}X_{i+1}, X_n\text{]} \end{cases}$$



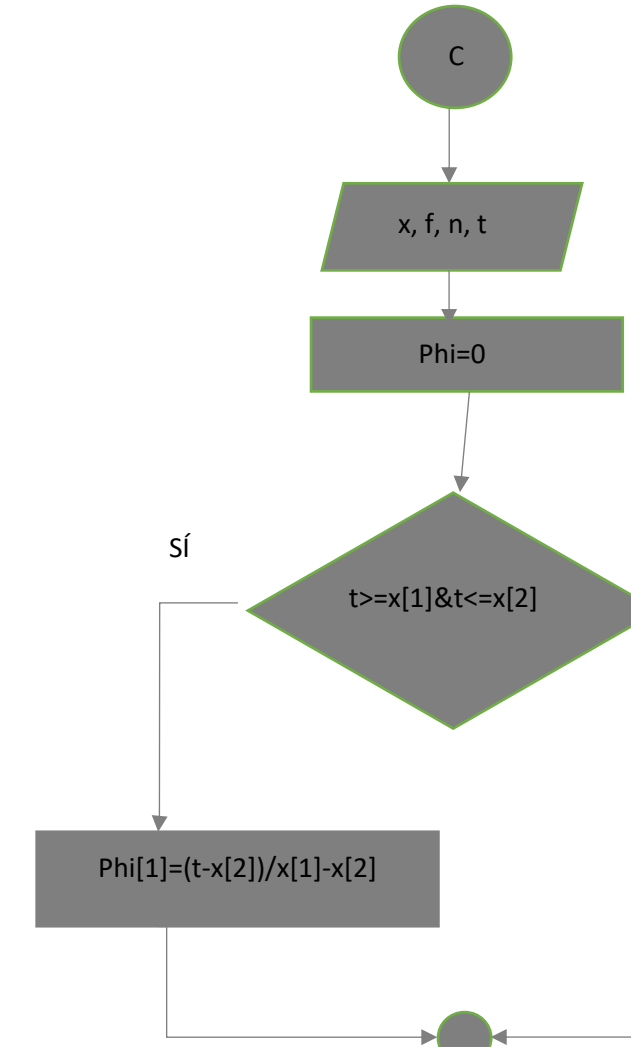
$$\varphi_n= \begin{cases} \frac{x-xn-1}{(xn-xn-1)} & \text{Si X}\varepsilon \text{ [}X_{n-1}, X_n\text{]} \\ 0 & \text{Si X}\varepsilon \text{ [}X_o, X_{n-1}\text{]} \end{cases}$$



ALGORITMO

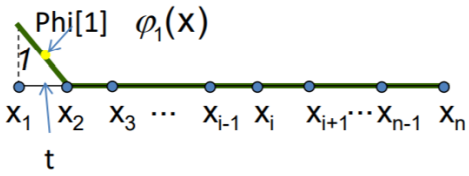
Comenzamos el algoritmo como siempre.

Los datos de entrada serán los puntos de la función original de los que conocemos sus imágenes (vector x); los valores de los puntos (vector f); el número de puntos que tenemos (n) y el punto t, que es el del que queremos conocer su valor mediante la interpolación.

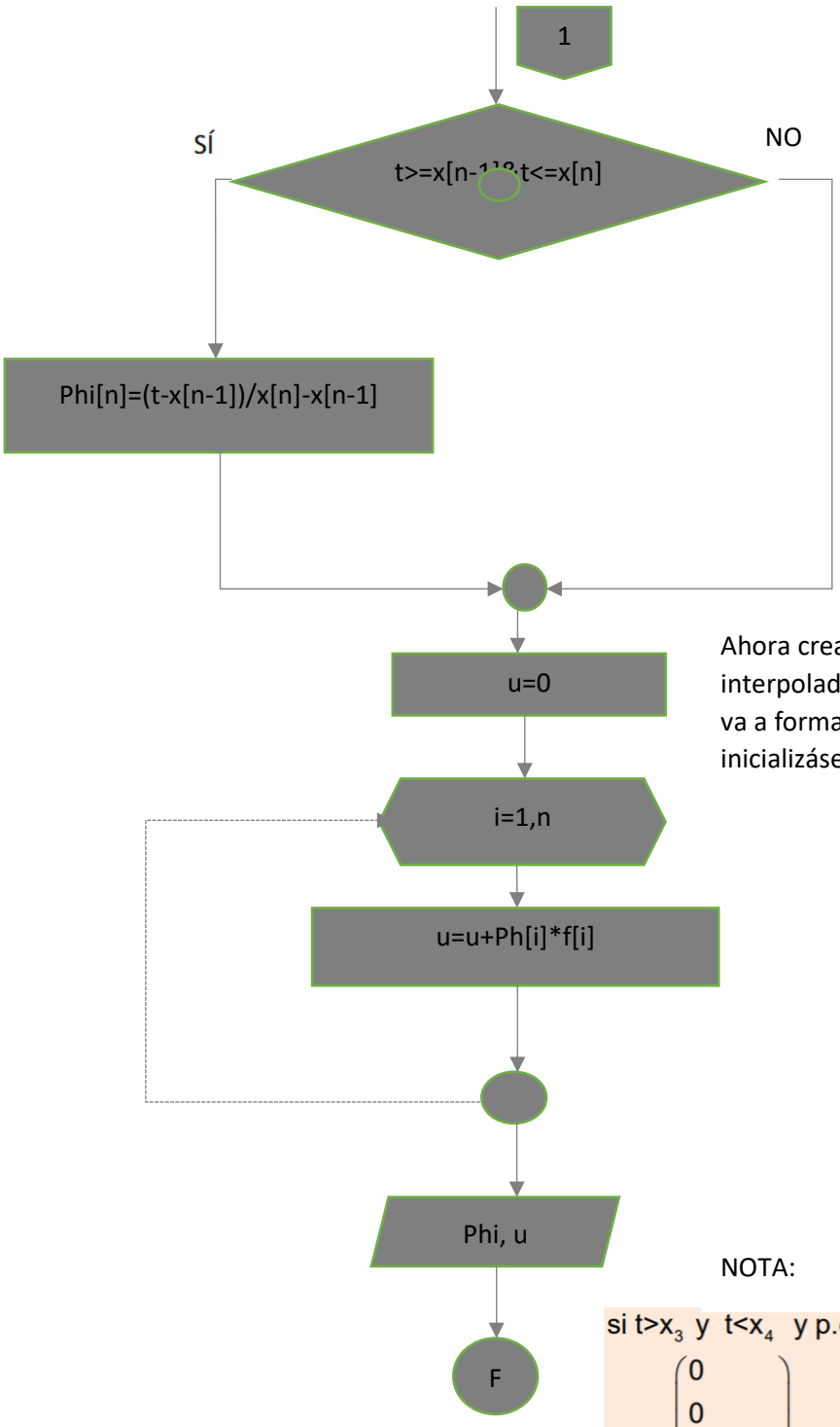


Inicializamos Phi (que representa el polinomio de base cuya fórmula general ha sido calculada anteriormente) en 0

Abrimos un bucle condicional y solo si t está en el intervalo del primer punto conocido y del segundo haremos su base de Lagrange.



Abrimos un bucle desde 2 hasta n-1, ya que ya hicimos el intervalo de los 2 primeros números. Entonces abrimos una condición, en la que si se cumple volveremos a obtener una base. En caso negativo abrimos otra condición para saber si la pendiente es positiva o negativa.



El último intervalo, igual que el primero hay que estudiarlo a parte

Ahora creamos el "polinomio interpolador" y lo inicializamos a 0 porque va a formar parte de un sumatorio. Si no lo inicializásemos a 0, el valor se perdería.

NOTA:

si $t>x_3$ y $t<x_4$ y p.e. tenemos n=7

$$Phi = \begin{pmatrix} 0 \\ 0 \\ \frac{t-x_4}{x_3-x_4} \\ \frac{t-x_3}{x_4-x_3} \\ 0 \\ 0 \\ 0 \end{pmatrix}$$