INTERPOLACIÓN DE LAGRANGE POR TRAMOS II

En este recurso vamos a resolver el problema de la Parte 1 de **Interpolación por Tramos**, utilizando la **fórmula de Newton**.

A continuación realizaremos el **Organigrama y Pseudo-Código** para calcular las Funciones de Base y la Función Interpoladora.

Recordemos el ejercicio:

Determinar la función continua polinómica de primer grado a trozos que interpola a una función f(x) de la que se conocen los siguientes datos:

x	0	2	5	6
f(x)	3	6	10.5	24

Resultado del algoritmo: un vector phi de n componentes que tengan el valor de cada función de base en cierto punto t y una variable u que contenga el valor interpolado en ese punto t dado.

Serán datos: Soporte de interpolación: x= {x1, x2,…,xn} Valores de la función: f={f1, f2,…,fn} Punto de interpolación t.

FÓRMULA DE NEWTON

Recordamos la diferencia dividida de orden 1 de una función f en los puntos $\{x_i, x_{i+1}\}$:

$$f[x_i, x_{i+1}] = \frac{f(x_{i+1}) - f(x_i)}{x_{i+1} - x_i}$$

Aplicando la fórmula de Newton en **diferencias divididas**, utilizaremos:

$$P^{(i)}(x) = f(x_i) + f[x_i, x_{i+1}] = \frac{f(x_{i+1}) - f(x_i)}{x_{i+1} - x_i}(x - x_i)$$

Así, en el caso que nos ocupa, obtendríamos:

$$P^{(1)}(x) = 3 + \frac{6-3}{2-0}(x-0) = 3 + \frac{3}{2}x$$

$$P^{(2)}(x) = 6 + \frac{10.5-6}{5-2}(x-2) = 3 + \frac{3}{2}x$$

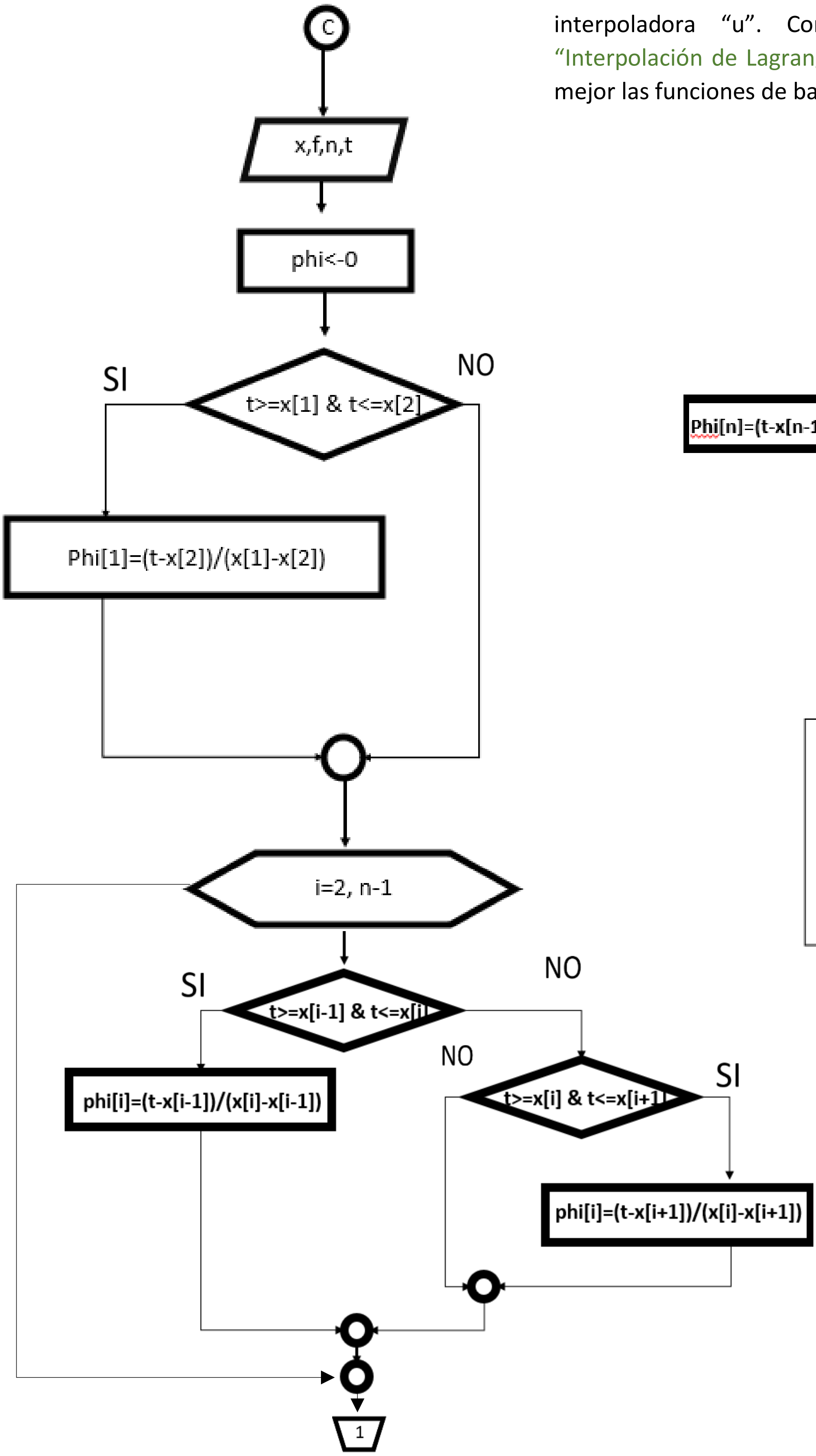
$$P^{(3)}(x) = 10.5 + \frac{24-10.5}{6-5}(x-5) = -57 + 13.5x$$

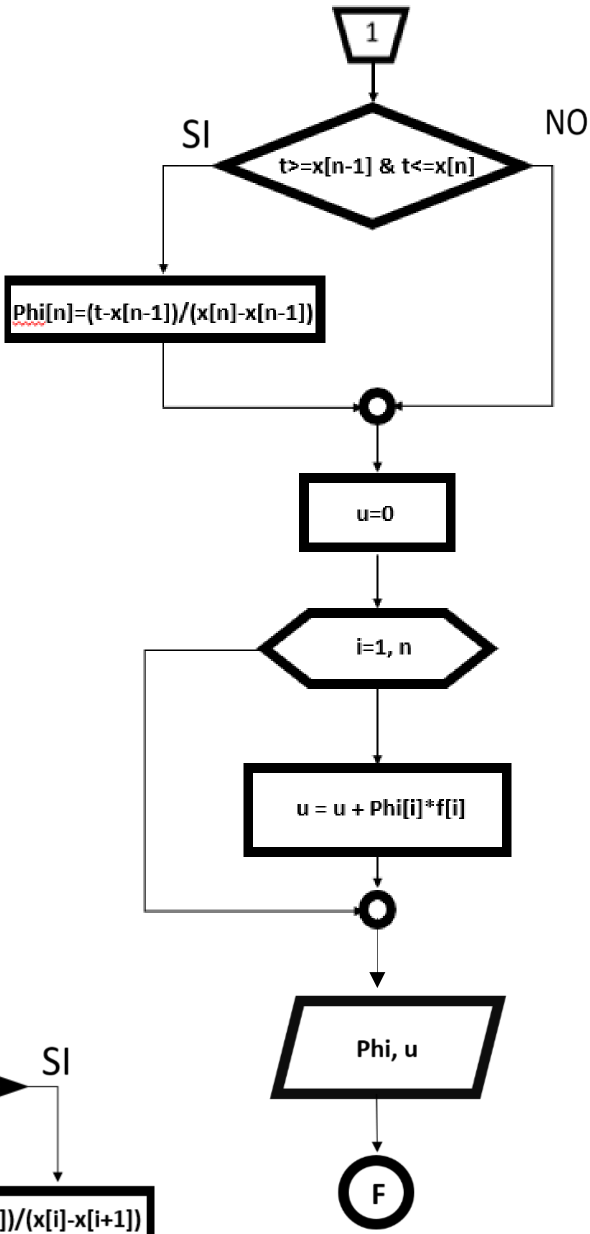
Una vez más, juntando los 3 polinomios obtenemos la función interpoladora u(x):

$$u(x) = \begin{cases} 3 + \frac{3}{2}x & \text{si } x \text{ en } [0,2] \\ 3 + \frac{3}{2}x & \text{si } x \text{ en } [2,5] \\ -57 + 13.5x & \text{si } x \text{ en } [5,6] \end{cases}$$

ORGANIGRAMA

A continuación realizamos el organigrama genérico para calcular las funciones de base (Phi) hasta un número "n" de funciones, además de la función interpoladora "u". Conviene revisar los apuntes "Interpolación de Lagrange por Tramos" para entender mejor las funciones de base.





PSEUDO-CÓDIGO

El pseudo-código nos ayuda a comprender mejor los pasos que damos en las diferentes partes del diagrama de flujo.

```
Inicio pseudo-código
            Leer x, f, n, t
             phi<-0
            Si (t>=x[1] & t <=x[2])
                phi[1]=(t-x[2])/(x[1]-x[2])
             Fin condición
             Para i desde 2 hasta (n-1)
                 Si (t>=x[i-1] & t<= x[i])
                     phi[i] ]=(t-x[i-1])/(x[i]-x[i-1])
                 Si no, si (t>=x[i] & t<= x[i+1])
                     phi[i]=(t-x[i+1])/(x[i]-x[i+1])
                 Fin condición
            Fin bucle
             Si (t>=x[n-1] & t<=x[n])
                    phi[n]=(t-x[n-1])/(x[n]-x[n-1])
            Fin condición
            u=0
            Para i desde 1 hasta n
                  u = u + f[i]*phi[i]
            Fin condición
            Escribir u, phi
Fin pseudo-código
```