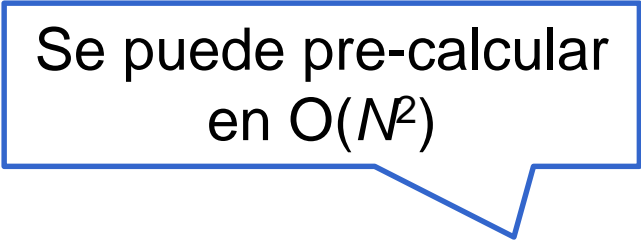


Árboles binarios de búsqueda óptimos, parte 2

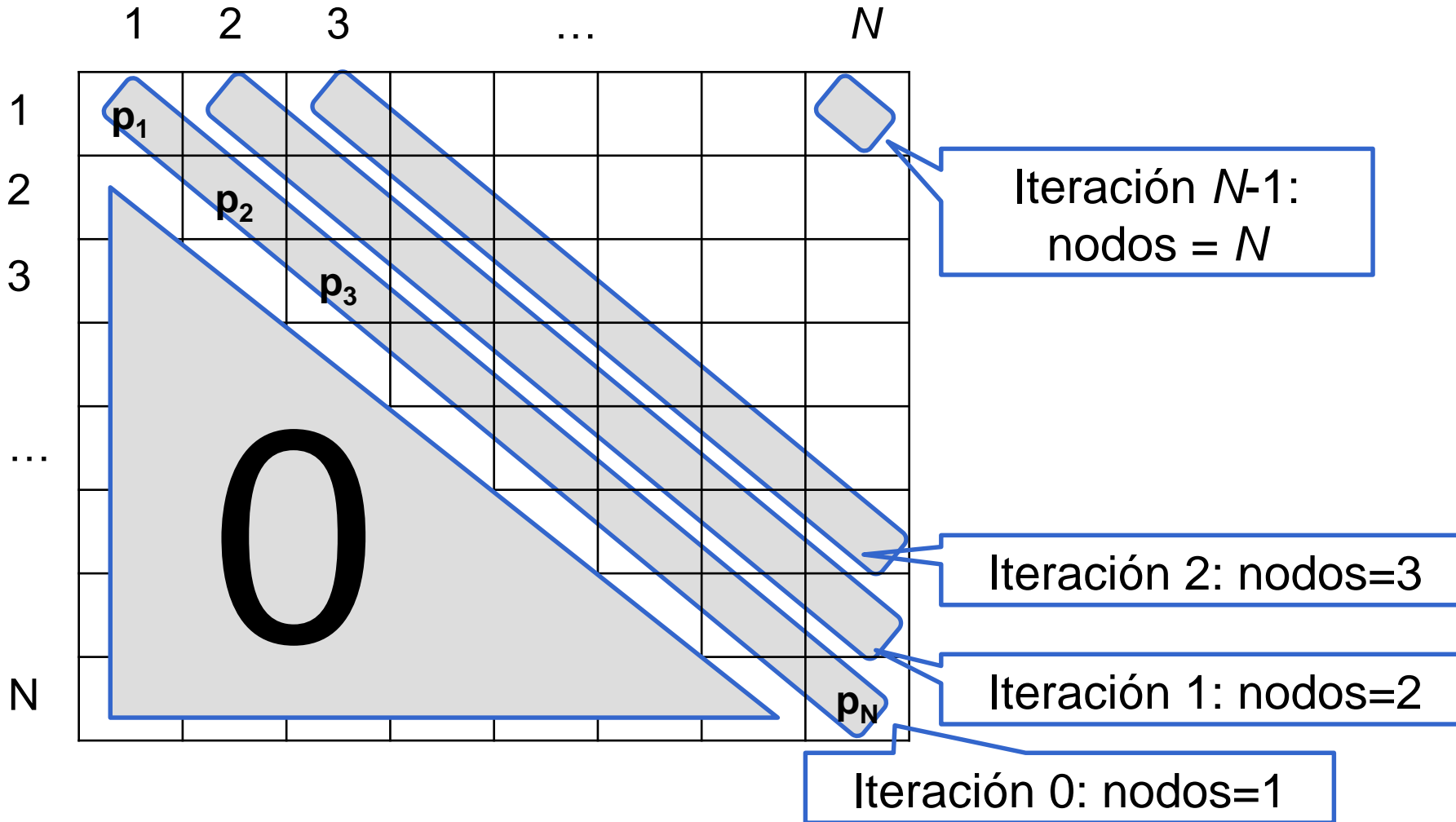
Solución mediante programación dinámica

```
for i = 2 to N:
    for j = 1 to i-1:
         $C_{i,j} = 0$ 
for i = 1 to N:
     $C_{i,i} = p_i$ 
for nodos = 2 to N:
    for i = 1 to N-nodos+1:
        j = i+nodos-1
        menor = INF
        for r = i to j:
            menor = MIN(menor,  $C_{i,r-1} + C_{r+1,j} + \text{sum}P_{i,j}$ )
         $C_{i,j} = \text{menor}$ 
print( $C_{1,N}$ )
```



Se puede pre-calcular
en $O(N^2)$

Solución mediante programación dinámica



¿Cuál es la complejidad de este algoritmo? $O(N^3)$

Solución mediante programación dinámica

Ejemplo: $P = \{A:30, B:10, C:25, D:35\}$

sumP

	1	2	3	4
1	30	40	65	100
2	0	10	35	70
3	0	0	25	60
4	0	0	0	35

C

	1	2	3	4
1	30	50	120	185
2	0	10	45	115
3	0	0	25	85
4	0	0	0	35

