# Análisis formal de complejidad para `dijkstra_tiempo`

Sean:

- $L =$ `MAX_LOCALIDADES` (número de localidades)

**1. Planteamiento y desarrollo de $T(L)$**
El algoritmo realiza:

1. Inicialización de un arreglo booleano y otro flotante de tamaño $L$: costo $t_1 + t_2 L$.
2. Un ciclo externo de $L - 1$ iteraciones:
   - En cada iteración, un ciclo interno de $L$ para buscar el nodo no visitado con menor distancia: $t_3 L$
   - Si se encuentra $u$, se marca como visitado: $t_4$
   - Otro ciclo de $L$ para actualizar distancias a vecinos: $t_5 L$

La estructura es:

$$T(L) = t_1 + t_2 L + \sum_{i=0}^{L-2} [t_3 L + t_4 + t_5 L]$$

Resolviendo la sumatoria:

$$T(L) = t_1 + t_2 L + (L-1)\,[t_3 L + t_4 + t_5 L]$$

$$T(L) = t_1 + t_2 L + (L-1)t_3 L + (L-1)t_4 + (L-1)t_5 L$$

$$T(L) = t_1 + t_2 L + (t_3 + t_5)L(L-1) + (L-1)t_4$$

Redefiniendo $c_1 = t_3 + t_5$, $c_2 = t_1 + t_2 L + (L-1)t_4$:

$$T(L) = c_1 L(L-1) + c_2$$

$$T(L) = c_1(L^2 - L) + c_2$$

Para cotas asintóticas, el término dominante es $L^2$, así que:

$$T(L) \sim c_1 L^2 \text{ cuando } L \to \infty$$

---

**2. Cotas de complejidad y comprobación con límites**
**Mejor caso:** El algoritmo siempre recorre ambos ciclos internos completos (no depende de los datos). Así que:

$$T_m(L) = c_1 L^2 + c_2$$

**Peor caso:** Igual que el mejor caso, ya que la estructura de los ciclos no depende del grafo.

$$T_p(L) = c_1 L^2 + c_2$$

**Caso promedio:** El costo también es idéntico (ya que todos los ciclos se recorren completos):

$$T_{pr}(L) = c_1 L^2 + c_2$$

**Límites:**

$$\lim_{L \to \infty} \frac{T(L)}{L^2} = c_1$$

Por lo tanto:

$$T(L) \in \mathcal{O}(L^2)$$

$$T(L) \in \Omega(L^2)$$

$$T(L) \in \Theta(L^2)$$

**3. Resumen Final:**

- Mejor caso: $\Theta(L^2)$
- Peor caso: $\Theta(L^2)$
- Caso promedio: $\Theta(L^2)$

Cada cota está verificada con límites al infinito, usando los nombres de variables reales del código.