ALGORITMIA

Tema 1:

Análisis de Algoritmos

Grado en Ingeniería Informática en Tecnologías de la Información
Escuela Politécnica de Ingeniería – Campus de Gijón
Universidad de Oviedo
2022 – 2023

Presentación

Este boletín contiene la solución a una veintena de problemas, cuyos enunciados se encuentran en el documento “Enunciados Problemas Tema1”. El documento “Soluciones Problemas Tema1” contiene los enunciados y la solución final, sin detallar los cálculos, y, para algunos problemas, consideraciones adicionales (p. ej. figuras o gráficos). Por tanto, todos ellos se complementan.

Estos problemas tratan de cubrir toda la casuística que el alumno de Algoritmia se encontrará al estudiar la complejidad temporal de los algoritmos que desarrolle en la asignatura (y en el resto de las asignaturas de la titulación). Grosso modo:

- Algoritmos Iterativos. Algo más de la mitad son algoritmos iterativos, tal que
  - Los problemas 3 al 9 no presentan mejor/peor.
  - Los problemas 2, 10, 18 y 19 presentan mejor y peor caso.
  - Los problemas 7, 18 y 19 presentan reducciones logarítmicas.
  - El problema 2 usa acotaciones.
  - Los problemas 8 y 9 usan funciones auxiliares.

- Algoritmos Recursivos
  - Los problemas 12, 13 y 15 no presentan mejor/peor.
  - Los problemas 11, 14 y 16 presentan mejor y peor caso.
  - Los problemas 15 y 17 usa acotaciones.
  - Los problemas 12 y 13 usan funciones auxiliares.

Finalmente, para simplificar los cálculos se va a suponer que todas las operaciones elementales o básicas tienen de coste “1 paso”, y que los “iteradores” (*for*, *while*, etc.), los selectores (*if*, *switch*, *case*, etc.), etc. sirven para definir el número de repeticiones de las operaciones básicas asociadas (establecen el rango de los $\sum$, $\prod$, etc.). Un análisis más detallado/preciso, como el usada en las clases de teoría, conducirá al mismo resultado en términos de notaciones asintóticas. En el problema “Búsqueda Secuencial”, el primero, se usan ambos enfoques para ilustrar el alcance del planteamiento seguido.

Búsqueda Secuencial

Talla: **n**
Mejor y peor caso: **Sí**

- Mejor caso ($A[1] = x$)

$$T_{MC}(n) = 1 + 2 + 1 \in \theta(1)$$

Esto es, "i=1" que es un paso, "retorna i" que es otro paso y 2 pasos por comprobar una sola vez las condiciones ($i \leq n$) y ($A[i] \neq x$).

- Peor caso ($(\forall i)\ (A[i] \neq x : 1 \leq i \leq n)$)

$$T_{PC}(n) = 1 + \sum_{i=1}^{n} (2 + 2) + 1 = 2 + 4n \in \theta(n)$$

que sigue la metodología de contar pasos usada en el mejor caso. Se está suponiendo que "i=i+1" son 2 pasos.

Por tanto,

$$T(n) \in \Omega(1) \wedge T(n) \in O(n)$$

Con este criterio, este ejercicio se podría haber resuelto como

- Mejor caso ($A[1] = x$)

$$T_{MC}(n) = 1 + 1 \in \theta(1)$$

- Peor caso ($(\forall i)\ (A[i] \neq x : 1 \leq i \leq n)$)

$$T_{PC}(n) = 1 + \sum_{i=1}^{n} (1) + 1 = 2 + n \in \theta(n)$$

Ejemplo 1

Procedimiento DosA: No presenta mejor y peor caso. La pregunta ($\boldsymbol{n \geq 2}$) no determina ni mejor ni peor caso dado que está relacionada con la talla del problema. El hecho de que $\boldsymbol{n}$ sea menor que 2 es irrelevante de cara a la eficiencia, pues ésta es una propiedad de carácter asintótico.

Procedimiento DosB: Sí presenta mejor y peor caso. La pregunta ($A[1] \geq 2$) determina mejor y peor caso dado que depende de la naturaleza de los datos de entrada y provoca un comportamiento distinto del algoritmo en el caso de que $A[1]$ sea mayor o igual que 2 y en el caso de que $A[1]$ sea menor que 2.

Ejemplo 2

No presenta mejor y peor caso. Para una **talla fija** ($n = valor$) el algoritmo siempre hace lo mismo. No obstante, según la paridad de la talla hace una cosa u otra, por lo que se debe estudiar por separado (acotar) la complejidad en las situaciones posibles. En este problema:

- Cuando n es par

$$T(n) = \sum_{i=1}^{n} 1 = n \in \theta(n)$$

- Cuando n es impar

$$T(n) = 1 \in \theta(1)$$

Luego

$$T(n) \in \Omega(1) \wedge T(n) \in O(n)$$

Ejemplo 3

Talla: **n**
Mejor y peor caso: **No**

$$T(n) = 2 + \sum_{i=1}^{n} 1 = 2 + n \in \theta(n)$$

Ejemplo 4

Talla: **(n, m)**
Mejor y peor caso: **No**

$$T(n,m) = \sum_{i=1}^{n} \left(\sum_{j=1}^{m} 1 \right) = \sum_{i=1}^{n} m = nm \in \theta(nm)$$

Ejemplo 5

Talla: **n**
Mejor y peor caso: **No**

$$T(n) = 4 + \sum_{i=1}^{n} 1 + \sum_{i=1}^{n} 2 = 4 + n + 2n \in \theta(n)$$

Ejemplo 6

Talla: **n**
Mejor y peor caso: **No**

$$T(n) = 2 + \sum_{i=1}^{n} \left(2 + \sum_{j=1}^{n} 2 \right) = 2 + \sum_{i=1}^{n} 2 + \sum_{i=1}^{n} 2n = 2 + 2n + 2n^2 \in \theta(n^2)$$

Ejemplo 7

Talla: **n**
Mejor y peor caso: **No**
El bucle realiza $1 + log2\ n$ iteraciones. Sabemos que la base del logaritmo no influye con respecto a su orden, luego:

$$T(n) = 2 + \sum_{i=0}^{log_2 n} 2 = 2 + 2(log_2 n + 1) = 4 + 2\, log_2 n \in \theta(log_2 n)$$

Ejemplo 8

Talla: **n**
Mejor y peor caso: **No**

$$T(n) = \sum_{i=1}^{n}(2+n) = \sum_{i=1}^{n} 2 + \sum_{i=1}^{n} n = 2n + n^2 \in \theta(n^2)$$

Ejemplo 9

Talla: **n**
Mejor y peor caso: **No**

$$T(n) = \sum_{i=1}^{n}(2+i) = \sum_{i=1}^{n} 2 + \sum_{i=1}^{n} i = 2n + \frac{n(n+1)}{2} = \frac{5n+n^2}{2} \in \theta(n^2)$$

Ejemplo 10

Talla: **n**
Mejor y peor caso: **Sí**

- Mejor caso $((\forall i)(A[i] \geq A[pos]: 1 \leq i \leq n))$

$$T_{MC}(n) = 2 + \sum_{1}^{n} 1 = 2 + n \in \theta(n)$$

- Peor caso $((\forall i)(A[i] < A[pos]: 1 \leq i \leq n))$

$$T_{PC}(n) = 2 + \sum_{i=1}^{n} 1 = 2 + n \in \theta(n)$$

Por tanto,

$$T(n) \in \Omega(n) \wedge T(n) \in O(n) \Rightarrow T(n) \in \theta(n)$$

Ejemplo 11

Talla: **n**
Mejor y peor caso: **Sí**

- Mejor caso $(A[n] = x)$

$$T_{MC}(n) = \begin{cases} c_1 & si\ (n = 0) \\ c_2 & si\ (n > 0) \end{cases} \quad \Rightarrow \quad T_{MC}(n) \in \theta(1)$$

- Peor caso $((\forall i)(A[i] \neq x: 1 \leq i \leq n))$

$$T_{PC}(n) = \begin{cases} c_1 & si\ (n = 0) \\ T_{PC}(n-1) + c_2 & si\ (n > 0) \end{cases}$$

$T_{PC}(n) = T_{PC}(n-1) + c_2 = T_{PC}(n-2) + 2c_2 = \cdots = T_{PC}(n-i) + ic_2$. La base se alcanza cuando $(n-i) = 0$, esto es, cuando $i = n$. Sustituyendo,

$$T_{PC}(n) = \cdots = T_{PC}(n-n) + nc_2 = T_{PC}(0) + nc_2 = c_1 + nc_2 \in \theta(n)$$

Por tanto,

$$T(n) \in \Omega(1) \wedge T(n) \in O(n)$$

Ejemplo 12

Talla: **n**
Mejor y peor caso: **No**

$$T(n) = \begin{cases} c_1 & si\ (n=1) \\ 2T(n-1) + c_2 & si\ (n>1) \end{cases}$$

$$\begin{aligned} T(n) &= 2T(n-1) + c_2 = \\ &= 2[2T(n-2) + c_2] + c_2 = 2^2T(n-2) + 2c_2 + c_2 = 2^2T(n-2) + (2^2-1)c_2 = \\ &= 2^2[2T(n-3) + c_2] + (2^2-1)c_2 = 2^3T(n-3) + 2^2c_2 + (2^2-1)c_2 = 2^3T(n-3) + (2^3-1)c_2 = \\ &\ldots \\ &= 2^iT(n-i) + (2^i-1)c_2 \end{aligned}$$

La base se alcanza cuando $(n-i) = 1$, por lo que $i = n-1$. En consecuencia,

$$T(n) = 2^{(n-1)}T(1) + \left(2^{(n-1)} - 1\right)c_2 \approx 2^nc_1 + 2^nc_2 \in \theta(2^n)$$

Ejemplo 13

Talla: $\boldsymbol{j-i+1}$. Por simplicidad, la denominaremos **n**.
Mejor y peor caso: **No**

$$T(n) = \begin{cases} c_1 & si\ (n \le 1)\ \ (\equiv i \ge j) \\ 2T(n\ div\ 2) + nc_2 + c_3 & si\ (n>1)\ \ (\equiv i < j) \end{cases}$$

$$\begin{aligned} T(n) &= 2T(n\ div\ 2) + nc_2 + c_3 = \\ &= 2[2T(n\ div\ 2^2) + (n\ div\ 2)c_2 + c_3] + nc_2 + c_3 = \\ &= 2^2T(n\ div\ 2^2) + 2(n\ div\ 2)c_2 + nc_2 + 3c_3 = \end{aligned}$$

$= 2^2T(n\ div\ 2^2) + 2nc_2 + 3c_3$ si n es par

$= 2^2T(n\ div\ 2^2) + (n-1)c_2 + nc_2 + 3c_3$ si n es impar

Acotando

$$\begin{aligned} &\approx 2^2T(n\ div\ 2^2) + 2nc_2 + 3c_3 \\ &= 2^2[2T(n\ div\ 2^3) + (n\ div\ 2^2)c_2 + c_3] + 2nc_2 + 3c_3 = \\ &\approx 2^3T(n\ div\ 2^3) + 3nc_2 + 7c_3 \end{aligned}$$

En la iteración *i-esima*

$$\approx 2^iT\left(n\ div\ 2^i\right) + inc_2 + \left(2^i - 1\right)c_3$$

Alcanzaríamos la base cuando $\left(n\ div\ 2^i\right) \le 1$ y esto sucede cuando $i \approx \log_2 n$. Reemplazamos i por $\log_2 n$ en la expresión anterior y tendremos

$$\approx 2^{\log_2(n)}c_1 + n\log_2(n)\,c_2 + \left(2^{\log_2(n)} - 1\right)c_3 = nc_1 + nc_2\log_2 n + (n-1)c_3 \in \theta(n\log_2 n)$$

Ejemplo 14

Talla: $(\boldsymbol{j - i + 1})$. Por simplicidad, la denominaremos **n**.
Mejor y peor caso: **Sí**

Mejor y peor caso: **Sí**

- Mejor caso ($A[1] \neq A[n]$)

$$T_{MC}(n) = \begin{cases} c_1 & si\ (n \leq 1) \\ c_2 & si\ (n > 1) \end{cases} \quad \Longrightarrow \quad T_{MC}(n) \in \theta(1)$$

- Peor caso ($(\forall(i,j)(A[i] = A[j])$)

$$T_{PC}(n) = \begin{cases} c_1 & si\ (n \leq 1) \\ T_{PC}(n-2) + c_2 & si\ (n > 1) \end{cases}$$

$T_{PC}(n) = T_{PC}(n-2) + c_2 = T_{PC}(n-4) + 2c_2 = \cdots = T_{PC}(n-2i) + ic_2$. La base se alcanza cuando $(n - 2i) = 1$, esto es, cuando $i = n\ div\ 2$. Sustituyendo,

$$T_{PC}(n) = \cdots = c_1 + (n\ div\ 2)c_2 \in \theta(n)$$

Por tanto,

$$T(n) \in \Omega(1) \wedge T(n) \in O(n)$$

Ejemplo 15

Talla: **n**
Mejor y peor caso: **No**

$$T(n) = \begin{cases} c_1 & si\ (n \leq 2) \\ T(n-1) + T(n-2) + c_2 & si\ (n > 2) \end{cases}$$

En la expresión anterior aumenta el número de términos según avanzamos hacia la base, siendo difícil establecer alguna fórmula general. Para solventar este problema podemos acotar $T(n)$ con las ecuaciones $T_1(n)$ y $T_2(n)$ siguientes:

$$T_1(n) = \begin{cases} c_1 & si\ (n \leq 2) \\ 2T_1(n-2) + c_2 & si\ (n > 2) \end{cases} \quad y \quad T_2(n) = \begin{cases} c_1 & si\ (n \leq 2) \\ 2T_2(n-1) + c_2 & si\ (n > 2) \end{cases}$$

Las ecuaciones $T_2(n)$ ya ha sido resueltas en Ejemplo 12.

$$T_1(n) = 2T_1(n-2) + c_2 =$$
$$= 2[2T_1(n-4) + c_2] + c_2 = 4T_1(n-4) + 3c_2 =$$
$$= 4[2T_1(n-6) + c_2] + 3c_2 = 8T_1(n-6) + 7c_2 =$$

…

$$= 2^i T_1(n-2i) + \left(2^i - 1\right)c_2$$

La base se alcanza cuando $n - 2i = 2$, en consecuencia $i = \frac{n-2}{2} \approx \frac{n}{2}$ Sustituyendo,

$$T_1(n) = \cdots = 2^{\frac{n}{2}}c_1 + \left(2^{\frac{n}{2}} - 1\right)c_2 \approx 2^{\frac{n}{2}}c_1 + 2^{\frac{n}{2}}c_2 \in \theta\left(2^{\frac{n}{2}}\right)$$

Por tanto,

$$T(n) \in \Omega\left(2^{n/2}\right) \wedge T(n) \in O(2^n)$$

Ejemplo 16

Talla: $(\boldsymbol{j} - \boldsymbol{i} + \mathbf{1})$. Por simplicidad, la denominaremos **n**.
Mejor y peor caso: **Sí**

- Mejor caso ($A[(1+n)\ div\ 2] = x$)

$$T_{MC}(n) = \begin{cases} c_1 & si\ (n = 0) \\ c_2 & si\ (n > 0) \end{cases} \quad \Longrightarrow \quad T_{MC}(n) \in \theta(1)$$

- Peor caso ($(\forall m)(A[m] \neq x)$). Observar que en el peor caso, la siguiente llamada recursiva trata aproximadamente la mitad de la sección anterior, es decir, se pasa de la sección $A[i \ldots j]$ a la sección $A[i \ldots m-1]$ o a la sección $A[m+1 \ldots j]$ donde m es la posición del elemento central. En consecuencia,

$$T_{PC}(n) = \begin{cases} c_1 & si\ (n = 0) \\ T_{PC}(n\ div\ 2) + c_2 & si\ (n > 0) \end{cases}$$

$T_{PC}(n) = T_{PC}(n\ div\ 2) + c_2 = T_{PC}(n\ div\ 4) + 2c_2 = \cdots = T_{PC}\left(n\ div\ 2^i\right) + ic_2$. La base se alcanza cuando $\left(n\ div\ 2^i\right) = 0$, esto es, cuando (aproximadamente) $n = 2^i \Longrightarrow i = \log_2 n$. Sustituyendo,

$$T_{PC}(n) = \cdots = c_1 + \log_2(n)\, c_2 \in \theta(\log_2 n)$$

Por tanto,

$$T(n) \in \Omega(1) \wedge T(n) \in O(\log_2 n)$$

Ejemplo 17

Talla: **n**
Mejor y peor caso: **No**

$$T(n) = \begin{cases} c_1 & si\ (n \leq 1) \\ T(n-1) + T(n-2) + T(n\ div\ 2) + c_2 & si\ (n > 1) \end{cases}$$

En la expresión anterior aumenta el número de términos según avanzamos hacia la base, siendo difícil establecer alguna fórmula general. Acotamos $T(n)$ con las ecuaciones $T_1(n)$ y $T_2(n)$ siguientes:

$$T_1(n) = \begin{cases} c_1 & si\ (n \leq 1) \\ 3T_1(n\ div\ 2) + c_2 & si\ (n > 1) \end{cases} \quad y \quad T_2(n) = \begin{cases} c_1 & si\ (n \leq 1) \\ 3T_2(n-1) + c_2 & si\ (n > 1) \end{cases}$$

$$T_1(n) = 3T_1(n\ div\ 2) + c_2 =$$

$$= 3[3T_1(n\ div\ 4) + c_2] + c_2 = 9T_1(n\ div\ 4) + (3c_2 + c_2) =$$

$$= 9[3T_1(n\ div\ 8) + c_2] + (3c_2 + c_2) = 27T_1(n\ div\ 8) + (9c_2 + 3c_2 + c_2) =$$

…

$$= 3^i T_1\left(n\ div\ 2^i\right) + \sum_{j=0}^{i-1} 3^j c_2$$

La base se alcanza cuando $n\ div\ 2^i = 1$, en consecuencia $i = \log_2 n$ Sustituyendo,

$$T_1(n) = \cdots = 3^{\log_2 n} c_1 + \sum_{j=0}^{\log_2(n)-1} 3^j c_2$$

Aplicando cambio de base ($\log_b x = \frac{\log_c x}{\log_c b}$) se tiene que $3^{\log_2 n} = 3^{\frac{\log_3 n}{\log_3 2}} = n^{\frac{1}{\log_3 2}} = n^{1.58}$. También se puede enfocar como: $3^{\log_2 n} = \left(2^{\log_2 3}\right)^{\log_2 n} = 2^{(\log_2 n)(\log_2 3)} = n^{\log_2 3} = n^{1.58}$. Además, se sabe que $\sum_{i=1}^{n} r^i \approx r^n$, por lo que $\sum_{j=0}^{\log_2 n-1} 3^j c_2 = c_2\left(1 + 3^{\log_2 n-1}\right) \approx c_2 3^{\log_2 n} = c_2 n^{1.58}$. Por tanto,

Por tanto,

$$T_1(n) = \cdots = c_1 n^{1.58} + c_2 n^{1.58} \in \theta(n^{1.58})$$

Ahora $T_2(n)$

$$\begin{aligned} T_2(n) &= 3T_2(n-1) + c_2 = \\ &= 3[3T_2(n-2) + c_2] + c_2 = 9T_2(n-2) + (3c_2 + c_2) = \\ &= 9[3T_2(n-3) + c_2] + (3c_2 + c_2) = 27T_2(n-3) + (9c_2 + 3c_2 + c_2) = \\ &\ldots \\ &= 3^i T_2(n-i) + \sum_{j=0}^{i-1} 3^j c_2 \end{aligned}$$

La base se alcanza cuando $(n-i) = 1 \Longrightarrow i \approx n$, sustituyendo:

$$T_2(n) = \cdots = 3^n c_1 + \sum_{j=0}^{n-1} 3^j c_2 \approx 3^n c_1 + 3^n c_2 \in \theta(3^n)$$

Por tanto,

$$T(n) \in \Omega(n^{1.58}) \wedge T(n) \in O(3^n)$$

Ejemplo 18

Talla: **n** (número de elementos de la lista)

Mejor y peor caso: **Sí**. El número de veces que itera el bucle mientras es $(1 + \log n)$ y los valores que va tomando la variable m dentro del mismo son: $\{2^0, 2^1, \ldots, 2^i, \ldots, 2^{\log n}\}$. Por tanto:

- Mejor caso $((\forall m)\ (Am = 0))$

$$T_{MC}(n) = 3 + \sum_{i=0}^{\log n} 1 = 3 + 1 + \log n \in \theta(\log n)$$

- Peor caso $((\forall m)(A[m] \neq 0))$

$$T_{PC}(n) = 3 + \sum_{i=0}^{\log n}\left(1 + \sum_{j=m}^{n} 1\right) = 4 + \log n + \sum_{i=0}^{\log n}\sum_{j=2^i}^{n} 1 =$$

$$= 4 + \log n + \sum_{i=0}^{\log n}\left(n - 2^i + 1\right) = 4 + \log n + \sum_{i=0}^{\log n} n - \sum_{i=0}^{\log n} 2^i + \sum_{i=0}^{\log n} 1 =$$

$$= 4 + \log n + n(1 + \log n) - \left(1 + \sum_{i=1}^{\log n} 2^i\right) + (1 + \log n) =$$

$$= 5 + 2\log n + n + \text{n}\log n - \left(1 + 2^{\log n}\right) = 4 + 2\log n + \text{n}\log n \in \theta(n \log n)$$

Por tanto,

$$T(n) \in \Omega(\log n) \wedge T(n) \in O(n \log n)$$

Ejemplo 19

Talla: **n**
Mejor y peor caso: **Sí**

- Mejor caso ($A[(1 + n)\ div\ 2] = x$)

$$T_{MC}(n) = 5 \in \theta(1)$$

- Peor caso ($(\forall m)(A[m] \neq x)$)
Observar que, en el peor caso, cada iteración del bucle *mientras* trata aproximadamente la mitad de la sección del vector de la iteración anterior, es decir, se pasa de la sección $A[i \dots j]$ a la sección $A[i \dots m - 1]$ o a la sección $A[m + 1 \dots j]$ donde m es la posición del elemento central. Por tanto, dado que el bucle comienza con una sección a tratar de tamaño n, itera hasta que la sección a tratar sea vacía y dicha sección a tratar se va dividiendo reiteradamente por 2 ($n, n\ div\ 2, n\ div\ 2^2$, ...), el bucle mientras itera $1 + \log n$ veces.

$$T_{pC}(n) = 3 + \sum_{i=0}^{\log n} 2 = 3 + 2(1 + \log n) = 5 + 2\log n \in \theta(\log n)$$