Cours de l'option informatique

Lycée Louis-le-Grand, Paris

Année 2002–2003

Récurrences classiques

sommaire

— notations ;

— équivalents à connaître ;

— récurrence d'ordre 1 ;

— récurrence diviser pour régner.

Notations

Les fonctions considérées vont de $\mathbb{N}$ dans lui-même. On pourra les supposer toujours strictement positives.

Toutes les estimations ont lieu quand la variable n tend vers l'infini.

Cela permet de simplifier la définition de la notation O qui devient

Définition 1 (notation O) On note $f(n) = O(g(n))$ s'il existe une constante $C > 0$ telle que $\forall n, f(n) \leqslant Cg(n)$.

On rappelle la définition mathématique de l'équivalent :

Définition 2 (équivalent) On note $f(n) \sim g(n)$ si le rapport $(f(n)/g(n))$ converge de limite 1.

En informatique, on préfère le plus souvent utiliser la notation Θ (d'aucuns préfèrent Ω) qui correspond à la notion d'*ordre de grandeur*.

On la définit ainsi

Définition 3 (notation Θ) On note $f(n) = \Theta(g(n))$ s'il existe deux constantes $A > 0$ et $B > 0$ telles que $\forall n, Ag(n) \leqslant f(n) \leqslant Bg(n)$.

On dit alors que $f(n)$ et $g(n)$ sont du même ordre de grandeur, il s'agit d'une relation d'équivalence.

Autrement dit, on a $f(n) = \Theta(g(n))$ si on a à la fois $f(n) = O(g(n))$ et $g(n) = O(f(n))$.

On remarquera que pour la plupart des fonctions $n \mapsto f(n)$ qui interviennent classiquement dans les récurrences utiles en dénombrement il existe une fonction $n \mapsto g(n)$ croissante telle que $f(n) = \Theta(g(n))$.

Quelques équivalents classiques

Théorème 1

On dispose, pour $\alpha > 0$ et $\omega > 1$, de :

$$\sum_{k=1}^{n} \omega^k = \Theta(\omega^n) \quad \text{et} \quad \sum_{k=1}^{n} k^\alpha = \Theta(n^{\alpha+1});$$

$$\begin{aligned}
\sum_{k=1}^{n} k^\alpha \lg^\beta k &= \Theta(n^{\alpha+1} \lg^\beta n); \\
\sum_{k=1}^{n} \omega^k k^\alpha \lg^\beta k &= \Theta(\omega^n n^\alpha \lg^\beta n); \\
\sum_{k=1}^{n} \frac{k^\alpha \lg^\beta k}{\omega^k} &= \Theta(1).
\end{aligned}$$

Récurrence $T(n+1) = aT(n) + f(n)$

Une solution exacte, mais peu utile

Divisant par a^{n+1} et posant $U(n) = T(n)/a^n$, il vient $U(n+1) = U(n) + f(n)/a^n$, de sorte que la solution est clairement $U(n) = U(0) + \sum_{k=1}^{n} \frac{f(k)}{a^k}$ ou encore

$$T(n) = a^n T(0) + \sum_{k=1}^{n} f(k) a^{n-k}.$$

Toute la difficulté est bien entendu dans l'estimation de la somme qui figure ici.

Une remarque pertinente

Supposons qu'il existe une fonction g telle que $f(n) = \Theta(g(n))$. Soit alors U qui vérifie la récurrence $U(n+1) = aU(n) + g(n)$ avec $U(0) = T(0)$.

On montre alors que $T(n) = \Theta(U(n))$: il suffit de prouver que si $f(n) = O(g(n))$ alors $T(n) = O(U(n))$, ce qu'établit facilement une récurrence.

En pratique, on remplace $f(n)$ par une expression du même ordre de grandeur de la forme ω^n ou $n^\alpha \lg^\beta n$, et on résout d'abord la récurrence en U.

Récapitulation : récurrence $T(n+1) = aT(n) + f(n)$

	$\Theta(f(n))$	$\Theta(T(n))$
$a = 1$	ω^n	ω^n
	$n^\alpha \lg^\beta n$	$n^{\alpha+1} \lg^\beta n$
	$\omega^n n^\alpha \lg^\beta n$	$\omega^n n^\alpha \lg^\beta n$
$a > 1$	ω^n	$a^n, \quad \text{si } \omega < a$
		$na^n, \quad \text{si } \omega = a$
		$\omega^n, \quad \text{si } \omega > a$
	$n^\alpha \lg^\beta n$	a^n
	$\omega^n n^\alpha \lg^\beta n$	$a^n, \quad \text{si } \omega < a$
		$a^n n^{\alpha+1} \lg^\beta n, \quad \text{si } \omega = a$
		$\omega^n n^\alpha \lg^\beta n, \quad \text{si } \omega > a$

Récurrence

$$T(n) = aT(\lfloor n/2 \rfloor) + bT(\lceil n/2 \rceil) + f(n)$$

Il s'agit ici de la récurrence habituelle qui correspond aux algorithmes qui suivent le paradigme *diviser pour régner*.

Par exemple, dans le cas du tri-fusion, on a $a = b = 1$ et $f(n) = \Theta(n)$: pour trier, il faut d'abord partager la liste en deux (coût linéaire, à compter dans $f(n)$), puis trier chaque moitié (même inexactes) (c'est donc ici $a = b = 1$) et enfin fusionner (coût linéaire, deuxième contribution à $f(n)$).

Dans toute la suite, nous supposerons qu'on a toujours $a \geqslant 0$, $b \geqslant 0$ et $a + b \geqslant 1$ et on travaillera pour $n \geqslant 1$.

Ordre de grandeur

Théorème 2

Si on dispose de $f(n) = \Theta(g(n))$, alors les solutions T ((*resp.* S)) de la récurrence $T(n) = aT(\lfloor n/2 \rfloor) + bT(\lceil n/2 \rceil) + f(n)$ ((*resp.* $S(n) = aS(\lfloor n/2 \rfloor) + bS(\lceil n/2 \rceil) + g(n)$)) avec $T(1) = S(1)$ vérifient $T(n) = \Theta(S(n))$.

✧ Une simple récurrence montre que $T(n) = O(S(n))$ et réciproquement. ❖

En pratique, on commencera systématiquement par remplacer dans une récurrence de cette forme le terme "$f(n)$" par une expression "$g(n)$" qui renferme l'ordre de grandeur étudié. En pratique cela signifie aussi qu'on peut supposer $f(n)$ positive, croissante, de la forme $\omega^n n^\alpha \lg^\beta n$.

Croissance

Théorème 3

Si la fonction f est croissante et positive, alors la solution T de la récurrence $T(n) = aT(\lfloor n/2 \rfloor) + bT(\lceil n/2 \rceil) + f(n)$ est aussi croissante, si l'on suppose $a \geqslant 0$, $b \geqslant 0$ et $a + b \geqslant 1$.

✧ Supposant que $1 \leqslant p \leqslant q \leqslant n \Rightarrow T(p) \leqslant T(q)$ (ce qui est déjà vrai pour $n = 2$ car $T(2) \geqslant (a+b)T(1) \geqslant T(1)$), on écrit

$$\begin{array}{rcl} T(n) & = & aT(\lfloor n/2 \rfloor) + bT(\lceil n/2 \rceil) + f(n) \\ T(n+1) & = & aT(\lfloor (n+1)/2 \rfloor) + bT(\lceil (n+1)/2 \rceil) + f(n+1) \end{array}$$

et on conclut car f croît, et car $1 \leqslant \lfloor n/2 \rfloor \leqslant \lfloor (n+1)/2 \rfloor$ et $1 \leqslant \lceil n/2 \rceil \leqslant \lceil (n+1)/2 \rceil$. ✦

Cas où n est une puissance de 2

Notons $U(p) = T(2^p)$.

La suite $U(p)$ vérifie maintenant la récurrence $U(0) = T(1)$ puis $U(p+1) = (a+b)U(p) + f(2^p)$, qui est du type étudié plus haut.

Notons désormais $\varphi = \lg(a+b)$, où, si l'on préfère, $a+b = 2^\varphi$, avec $\varphi \geqslant 0$.

La récurrence se réécrit $U(p+1) = 2^\varphi U(p) + f(2^p)$ et se résout ainsi :

$$\begin{aligned} T(2^p) &= U(p) = 2^{p\varphi} \left(U(0) + \sum_{k=1}^{p} \frac{f(2^k)}{2^{k\varphi}} \right) \\ &= (a+b)^p \left(T(1) + \sum_{k=1}^{p} \frac{f(2^k)}{(a+b)^k} \right). \end{aligned}$$

Un exemple

Considérons l'exemple de la récurrence $T(n) = T(\lfloor n/2 \rfloor) + T(\lceil n/2 \rceil) + \Theta(n)$, ce qui correspond précisément au tri-fusion.

On a vu que tant qu'on ne s'intéresse qu'aux ordres de grandeur, on peut poser $f(n) = n$.

On sait alors que la solution est croissante, et que si $U(p) = T(2^p)$, $U(p)$ est solution de $U(p+1) = 2U(p) + 2^p$.

L'étude précédente a montré que $U(p) = \Theta(p2^p)$. Mais alors, pour tout entier n, grâce à la croissance de T, on peut écrire :

$$\Theta(n \lg n) = U(\lfloor \lg n \rfloor) \leqslant T(n) \leqslant U(\lceil \lg n \rceil) = \Theta(n \lg n),$$

et donc $T(n) = \Theta(n \lg n)$.

Récurrence $T(n) = aT(\lfloor n/2 \rfloor) + bT(\lceil n/2 \rceil) + f(n)$

On pose $\varphi = \lg(a+b)$ ou encore $a + b = 2^{\varphi}$, avec $a \geqslant 0$, $b \geqslant 0$ et $\varphi \geqslant 0$.

	$\Theta(f(n))$	$\Theta(T(n))$
$a + b = 1,\ \varphi = 0$	n^{α}	n^{α}
	$n^{\alpha} \lg^{\beta} n$	$n^{\alpha} \lg^{\beta} n$
$a + b > 1,\ \varphi > 0$	n^{α}	$(a+b)^{\lg n} = n^{\varphi}$, si $\alpha < \varphi$
		$n^{\varphi} \lg n$, si $\alpha = \varphi$
		n^{α}, si $\alpha > \varphi$
	$n^{\alpha} \lg^{\beta} n$	$(a+b)^{\lg n} = n^{\varphi}$, si $\alpha < \varphi$
		$n^{\varphi} \lg^{\beta+1} n$, si $\alpha = \varphi$
		$n^{\alpha} \lg^{\beta} n$, $\alpha > \varphi$