

Chap. 3 Récursion et induction.

Les définitions par récurrence consistent à construire des objets finis, à partir d'autres, selon certaines règles. Les démonstrations par récurrence permettent de raisonner sur les objets construits de cette façon.

L'informatique manipule précisément des objets de ce type. Les définitions récursives interviennent constamment dans les SDD, de même que dans la conception des programmes récursifs.

Raisonnement par récurrence sur \mathbf{N}

Premier principe d'induction.

Sur \mathbf{N} , le premier principe d'induction, aussi appelé souvent principe de récurrence mathématique, se révèle être un mode de raisonnement particulièrement utile.

Théorème Soit $P(n)$ une propriété dépendant de l'entier n . Si les deux conditions suivantes sont vérifiées :

(B) $P(0)$ est vrai,

(I) $\forall n \in \mathbf{N}, (P(n) \Rightarrow P(n+1))$,

alors $\forall n \in \mathbf{N}, P(n)$ est vrai.

(B) s'appelle l'étape de base de la récurrence (ou induction) et (I) l'étape inductive (on dit aussi parfois "passage de n à $n+1$ ").

Démonstration. On raisonne par l'absurde. On considère l'ensemble

$$X = \{k \in \mathbf{N} / P(k) \text{ est faux}\}.$$

Si X est non vide, il admet un plus petit élément n . D'après la condition (B) $n \neq 0$. Donc, $n - 1$ est un entier et $n - 1 \notin X$, c'est-à-dire $P(n - 1)$ est vrai. En utilisant (I), on obtient alors $P(n)$ vrai ce qui est une contradiction avec $n \in X$. Par conséquent, X est vide, ce qui prouve le théorème.

Exemple On cherche à calculer la somme $S_n = 1 + 2 + \dots + n$. On remarque que

$$2S_1 = 2 = 1 * 2, 2S_2 = 2 + 4 = 2 * 3, 2S_3 = 2 + 4 + 6 = 3 * 4.$$

On conjecture alors que pour $\forall n > 0, 2S_n = n(n+1)$. On va le montrer par récurrence.

Soit $P(n)$ la propriété " $2S_n = n(n+1)$ ", on vérifie

(B) $2S_1 = 1 * 2$

(I) Soit $n \geq 1$, supposons $P(n)$. On a

$$2S_{n+1} = 2S_n + 2(n+1) = n(n+1) + 2(n+1) = (n+1)(n+2),$$

donc $P(n+1)$ est vraie.

On peut alors en conclure que $\forall n \geq 1, P(n)$.

Deuxième principe d'induction.

Dans le premier principe d'induction, la véracité de $P(n + 1)$ dépend uniquement de celle de $P(n)$, c'est-à-dire si la proposition P est vraie à l'étape n elle est aussi vraie à l'étape $(n + 1)$. Il se peut qu'on ait des cas plus complexes, où pour établir que P est vraie à l'étape $(n+1)$ on ait besoin d'utiliser explicitement le fait que P est vraie aux étapes $0, 1, \dots, n - 1, n$. En ce cas il est plus pratique d'utiliser le deuxième principe d'induction qui s'énonce comme suit.

Théorème Soit $P(n)$ une propriété dépendant de l'entier n . Si la proposition suivante est vérifiée :

$$(I) \quad \forall n \in \mathbf{N}, ((\forall k < n, P(k)) \Rightarrow P(n))$$

alors $\forall n \in \mathbf{N}$, $P(n)$ est vraie.

Ensembles définis inductivement

Il est très fréquent en informatique de définir inductivement (récursivement) des parties d'ensemble. En particulier, bon nombre de structures de données peuvent être définies de la sorte. Intuitivement, la définition inductive d'une partie X d'un ensemble consiste en la donnée explicite de certains éléments de l'ensemble X et de moyens de construire de nouveaux éléments de X à partir d'éléments déjà connus. Une telle définition se présentera donc sous la forme générique intuitive suivante

(B) certains éléments de l'ensemble X sont donnés explicitement (bases de la définition récursive),

(I) les autres éléments de l'ensemble X sont définis en fonction d'éléments appartenant déjà à l'ensemble X (étapes inductives de la définition récursive). Formellement, nous avons

Définition Soit E un ensemble. Une définition inductive d'une partie X de E consiste en la donnée :

- d'un sous-ensemble B de E
- d'un ensemble K d'opérations $\Phi : E^{a(\Phi)} \rightarrow E$, où $a(\Phi) \in \mathbf{N}$ est l'arité de Φ .

X est défini comme étant le plus petit ensemble vérifiant les assertions (B) et (I) suivantes

(B) $B \subset X$,

(I) $\forall \Phi \in K, \forall x_1, \dots, x_{a(\Phi)} \in X, \Phi(x_1, \dots, x_{a(\Phi)}) \in X$.

Exemple

La partie X de \mathbf{N} définie inductivement par

(B) $0 \in X$

(I) $n \in X \Rightarrow n + 1 \in X$

n'est autre que \mathbf{N} tout entier. (B) et (I) constituent donc une définition inductive de \mathbf{N} .

Fonctions définies inductivement.

Lorsque' on veut définir une fonction sur un ensemble défini inductivement, il est très commode d'utiliser une définition inductive. Intuitivement, on définit la fonction directement sur les éléments de la base puis inductivement sur les nouveaux éléments construits à partir d'éléments déjà connus.

Définition Soit X un ensemble défini inductivement. La définition inductive d'une application f sur X consiste en

(B) la donnée de $f(x)$ pour chaque élément $x \in B$,

(I) l'expression de $f(\Phi((x_j)_{j \in J}))$ à partir des $(x_j)_{j \in J}$ et des $(f(x_j))_{j \in J}$ pour chaque $\Phi \in K$ d'arité $J = a(\Phi)$.

Cette définition un peu abstraite va s'éclairer sur un exemple.

Exemple

La fonction factorielle sur \mathbf{N} se définit inductivement par

(B) $\text{fact}(0) = 1$,

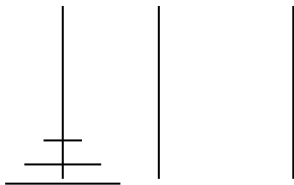
(I) $\text{fact}(n + 1) = (n+1) * \text{fact}(n)$.

On utilise ici la définition inductive de **N** donnée dans l'exemple précédent. On a bien exprimé la fonction factorielle directement pour l'unique élément de la base (0) et en fonction de n et $\text{fact}(n)$ pour le nouvel élément $n + 1$.

Exemple d'un problème récurrent.

Tours de Hanoi

Petit divertissement mathématique mis au point par un mathématicien français Edouard Lucas en 1883. Il consiste en 3 piquets, le premier porte n disques de tailles toutes différentes, empilés du plus grand (en bas) au plus petit (en haut). Le problème des tours de Hanoi consiste à faire passer tous ces disques au piquet 2, en s'aidant du troisième piquet, sachant qu'on ne déplace qu'un disque à la fois, et en respectant la règle suivante : aucun disque ne doit être empilé sur un disque de diamètre inférieur.



Certaines sources signalent l'arithméticien Edouard Lucas comme étant à l'origine de ce divertissement. D'autres en donnent une explication plus mythologique :

Dans le grand temple de Bénarès, sous le dôme qui marque le centre du monde, se trouve une plaque de bronze où sont fixées trois aiguilles de diamant, hautes chacune d'une coudée et fines comme la taille d'une guêpe. Sur une de ces aiguilles, lors de la création du monde, Dieu a placé 64 disques d'or pur, le plus large reposant sur la plaque de bronze, et les autres allant en décroissant jusqu'au plus petit. C'est la Tour de Bramah. Jour et nuit, sans arrêt, les prêtres transfèrent les disques d'une aiguille de diamant à une autre, en suivant les lois immuables de Bramah qui veulent que le prêtre de service ne prenne qu'un disque à la fois, et qu'il le place sur une aiguille de telle manière qu'il ne se trouve jamais sous lui de disque plus petit. Lorsque les soixante-quatre disques auront été transférés de l'aiguille sur laquelle Dieu les a mis lors de la création du monde, à une des autres aiguilles, la Tour, le Temple et les Brahmanes s'écrouleront en poussière, et dans un coup de tonnerre, le Monde s'évanouira.

Il existe un algorithme récursif très classique pour résoudre ce problème. Supposons qu'on sache déplacer $n-1$ disques. Pour en déplacer n , il suffit de déplacer $(n-1)$ disques du piquet 1 au piquet 3, puis de déplacer le grand disque du piquet 1 au piquet 2, on termine en déplaçant les $(n-1)$ autres disques du piquet 3 vers le piquet 2.

Si h_n est le nombre de déplacements de disques nécessaires, on a la formule de récurrence suivante : $h_n = 2h_{n-1} + 1$, avec la condition $h_1 = 1$. Par un calcul par récurrence facile, on trouve : $h_n = 2^n - 1$.

Démontrons par induction que $h_n = 2^n - 1$

(B) $h_1 = 1 = 2^1 - 1$

(I) Si $h_{n-1} = 2^{n-1} - 1$ alors

$$h_n = 2(2^{n-1} - 1) + 1 = 2^n - 1$$

La partie la plus difficile est la recherche de formule. Il faut commencer par deviner la solution !