De la combinatoire aux graphes (HLIN201) – L1

Définitions inductives et preuve par inductions structurelles

Sèverine Bérard

Université de Montpellier

2^e semestre 2017-18

Sommaire

Sommaire

Introduction

- Il est très fréquent en informatique de définir inductivement des ensembles
- En particulier, bon nombre de structures de données peuvent-être définies de la sorte.
- Intuitivement, la définition inductive d'une partie E d'un ensemble $\mathcal{U}$ consiste
  1. en la donnée explicite de certains éléments de l'ensemble E : *la base*
  2. et de moyens de construire de nouveaux éléments de E à partir d'éléments de E déjà connus : *les règles*
- Les règles sont des fonctions

Les définitions qui suivent peuvent paraître compliquées mais le principe est très simple, très intuitif !

Introduction

- Il est très fréquent en informatique de définir inductivement des ensembles
- En particulier, bon nombre de structures de données peuvent-être définies de la sorte.
- Intuitivement, la définition inductive d'une partie E d'un ensemble $\mathcal{U}$ consiste
  1. en la donnée explicite de certains éléments de l'ensemble E : *la base*
  2. et de moyens de construire de nouveaux éléments de E à partir d'éléments de E déjà connus : *les règles*
- Les règles sont des fonctions

Les définitions qui suivent peuvent paraître compliquées mais le principe est très simple, très intuitif !

Introduction

- Il est très fréquent en informatique de définir inductivement des ensembles
- En particulier, bon nombre de structures de données peuvent-être définies de la sorte.
- Intuitivement, la définition inductive d'une partie E d'un ensemble $\mathcal{U}$ consiste
  1. en la donnée explicite de certains éléments de l'ensemble E : *la base*
  2. et de moyens de construire de nouveaux éléments de E à partir d'éléments de E déjà connus : *les règles*
- Les règles sont des fonctions

Les définitions qui suivent peuvent paraître compliquées mais le principe est très simple, très intuitif !

Introduction

- Il est très fréquent en informatique de définir inductivement des ensembles

- En particulier, bon nombre de structures de données peuvent-être définies de la sorte.

- Intuitivement, la définition inductive d'une partie E d'un ensemble $\mathcal{U}$ consiste

  1. en la donnée explicite de certains éléments de l'ensemble E : *la base*
  2. et de moyens de construire de nouveaux éléments de E à partir d'éléments de E déjà connus : *les règles*

- Les règles sont des fonctions

Les définitions qui suivent peuvent paraître compliquées mais le principe est très simple, très intuitif !

Introduction

- Il est très fréquent en informatique de définir inductivement des ensembles
- En particulier, bon nombre de structures de données peuvent-être définies de la sorte.
- Intuitivement, la définition inductive d'une partie E d'un ensemble $\mathcal{U}$ consiste
  1. en la donnée explicite de certains éléments de l'ensemble E : *la base*
  2. et de moyens de construire de nouveaux éléments de E à partir d'éléments de E déjà connus : *les règles*
- Les règles sont des fonctions

Les définitions qui suivent peuvent paraître compliquées mais le principe est très simple, très intuitif !

Introduction

- Il est très fréquent en informatique de définir inductivement des ensembles

- En particulier, bon nombre de structures de données peuvent-être définies de la sorte.

- Intuitivement, la définition inductive d'une partie E d'un ensemble $\mathcal{U}$ consiste

  1. en la donnée explicite de certains éléments de l'ensemble E : *la base*

  2. et de moyens de construire de nouveaux éléments de E à partir d'éléments de E déjà connus : *les règles*

- Les règles sont des fonctions

Les définitions qui suivent peuvent paraître compliquées mais le principe est très simple, très intuitif !

Définition

Un ensemble $E \subseteq \mathcal{U}$ est défini inductivement

1. par la donnée de sa base : un ensemble $\mathcal{B} \subseteq \mathcal{U}$
2. et par un ensemble $\Omega = \Omega_i \cup \Omega_e$ de règles

Les règles

L'ensemble de règles peut être divisé en deux sous-ensembles (l'un des deux pouvant être vide)

1. Ω_i contenant les règles de construction internes
2. Ω_e contenant les règles de construction externes
   Pour ces dernières, on utilise des éléments d'autres ensembles définis par ailleurs

Définition

Un ensemble $E \subseteq \mathcal{U}$ est défini inductivement

1. par la donnée de sa base : un ensemble $\mathcal{B} \subseteq \mathcal{U}$
2. et par un ensemble $\Omega = \Omega_i \cup \Omega_e$ de règles

Les règles

L'ensemble de règles peut être divisé en deux sous-ensembles (l'un des deux pouvant être vide)

1. Ω_i contenant les règles de construction internes
2. Ω_e contenant les règles de construction externes
   Pour ces dernières, on utilise des éléments d'autres ensembles définis par ailleurs

Définition

Un ensemble $E \subseteq \mathcal{U}$ est défini inductivement

1. par la donnée de sa base : un ensemble $\mathcal{B} \subseteq \mathcal{U}$
2. et par un ensemble $\Omega = \Omega_i \cup \Omega_e$ de règles

Les règles

L'ensemble de règles peut être divisé en deux sous-ensembles (l'un des deux pouvant être vide)

1. Ω_i contenant les règles de construction internes
2. Ω_e contenant les règles de construction externes
   Pour ces dernières, on utilise des éléments d'autres ensembles définis par ailleurs

Définition

Un ensemble $E \subseteq \mathcal{U}$ est défini inductivement

1. par la donnée de sa base : un ensemble $\mathcal{B} \subseteq \mathcal{U}$
2. et par un ensemble $\Omega = \Omega_i \cup \Omega_e$ de règles

Les règles

L'ensemble de règles peut être divisé en deux sous-ensembles (l'un des deux pouvant être vide)

1. Ω_i contenant les règles de construction internes
2. Ω_e contenant les règles de construction externes
   Pour ces dernières, on utilise des éléments d'autres ensembles définis par ailleurs

Définition

Un ensemble $E \subseteq \mathcal{U}$ est défini inductivement

1. par la donnée de sa base : un ensemble $\mathcal{B} \subseteq \mathcal{U}$
2. et par un ensemble $\Omega = \Omega_i \cup \Omega_e$ de règles

Les règles

L'ensemble de règles peut être divisé en deux sous-ensembles (l'un des deux pouvant être vide)

1. Ω_i contenant les règles de construction internes
2. Ω_e contenant les règles de construction externes
   Pour ces dernières, on utilise des éléments d'autres ensembles définis par ailleurs

Plus formellement

Soient $K_1, \ldots, K_p \subseteq \mathcal{U}$ des ensembles parfaitement définis

Définition inductive d'un ensemble E

1. *(Base)* $\mathcal{B} \subseteq E$
2. *(Règles de construction internes)* Pour chaque règle $f_i \in \Omega_i$ d'arité n et pour chaque n-uplet $(x_1, \ldots, x_n) \in E^n$:
   $f_i(x_1, \ldots, x_n) \in E$
3. *(Règles de construction externes)* Pour chaque règle $f_e \in \Omega_e$ d'arité n définie de $K_1 \times \ldots \times K_p \times E \times \ldots \times E$ dans E ($p < n$) et pour chaque n-uplet $(x_1, \ldots, x_n)$ avec $x_1 \in K_1, \ldots, x_p \in K_p, x_{p+1} \in E, \ldots, x_n \in E$:
   $f_e(x_1, \ldots, x_n) \in E$

E est le plus petit ensemble vérifiant ces trois propriétés

Plus formellement

Soient $K_1, \ldots, K_p \subseteq \mathcal{U}$ des ensembles parfaitement définis

Définition inductive d'un ensemble E

1. *(Base)* $\mathcal{B} \subseteq E$
2. *(Règles de construction internes)* Pour chaque règle $f_i \in \Omega_i$ d'arité n et pour chaque n-uplet $(x_1, \ldots, x_n) \in E^n$:
   $f_i(x_1, \ldots, x_n) \in E$
3. *(Règles de construction externes)* Pour chaque règle $f_e \in \Omega_e$ d'arité n définie de $K_1 \times \ldots \times K_p \times E \times \ldots \times E$ dans E ($p < n$) et pour chaque n-uplet $(x_1, \ldots, x_n)$ avec $x_1 \in K_1, \ldots, x_p \in K_p, x_{p+1} \in E, \ldots, x_n \in E$:
   $f_e(x_1, \ldots, x_n) \in E$

E est le plus petit ensemble vérifiant ces trois propriétés

Plus formellement

Soient $K_1, \ldots, K_p \subseteq \mathcal{U}$ des ensembles parfaitement définis

Définition inductive d'un ensemble E

1. *(Base)* $\mathcal{B} \subseteq E$
2. *(Règles de construction internes)* Pour chaque règle $f_i \in \Omega_i$ d'arité n et pour chaque n-uplet $(x_1, \ldots, x_n) \in E^n$:
   $f_i(x_1, \ldots, x_n) \in E$
3. *(Règles de construction externes)* Pour chaque règle $f_e \in \Omega_e$ d'arité n définie de $K_1 \times \ldots \times K_p \times E \times \ldots \times E$ dans E ($p < n$) et pour chaque n-uplet $(x_1, \ldots, x_n)$ avec $x_1 \in K_1, \ldots, x_p \in K_p, x_{p+1} \in E, \ldots, x_n \in E$:
   $f_e(x_1, \ldots, x_n) \in E$

E est le plus petit ensemble vérifiant ces trois propriétés

Plus formellement

Soient $K_1, \dots, K_p \subseteq \mathcal{U}$ des ensembles parfaitement définis

Définition inductive d'un ensemble E

1. *(Base)* $\mathcal{B} \subseteq E$
2. *(Règles de construction internes)* Pour chaque règle $f_i \in \Omega_i$ d'arité n et pour chaque n-uplet $(x_1, \dots, x_n) \in E^n$:
   $f_i(x_1, \dots, x_n) \in E$
3. *(Règles de construction externes)* Pour chaque règle $f_e \in \Omega_e$ d'arité n définie de $K_1 \times \dots \times K_p \times E \times \dots \times E$ dans E ($p < n$) et pour chaque n-uplet $(x_1, \dots, x_n)$ avec $x_1 \in K_1, \dots, x_p \in K_p, x_{p+1} \in E, \dots, x_n \in E$:
   $f_e(x_1, \dots, x_n) \in E$

E est le plus petit ensemble vérifiant ces trois propriétés

Plus formellement

Soient $K_1, \ldots, K_p \subseteq \mathcal{U}$ des ensembles parfaitement définis

Définition inductive d'un ensemble E

1. *(Base)* $\mathcal{B} \subseteq E$
2. *(Règles de construction internes)* Pour chaque règle $f_i \in \Omega_i$ d'arité n et pour chaque n-uplet $(x_1, \ldots, x_n) \in E^n$:
   $f_i(x_1, \ldots, x_n) \in E$
3. *(Règles de construction externes)* Pour chaque règle $f_e \in \Omega_e$ d'arité n définie de $K_1 \times \ldots \times K_p \times E \times \ldots \times E$ dans E ($p < n$) et pour chaque n-uplet $(x_1, \ldots, x_n)$ avec $x_1 \in K_1, \ldots, x_p \in K_p, x_{p+1} \in E, \ldots, x_n \in E$:
   $f_e(x_1, \ldots, x_n) \in E$

E est le plus petit ensemble vérifiant ces trois propriétés

Exemple

L'ensemble *Pair* $\subseteq \mathbb{N}$

1. *(Base)* $\{0\}$
2. *(Règle)* si $p \in Pair$ alors $p + 2 \in Pair$

Remarques

- La base ne contient qu'un seul élément
- Il y a une seule règle, c'est une règle de construction interne
- que l'on pourrait définir plus formellement comme

$$\mathbf{r} : \begin{array}{ccc} Pair & \longrightarrow & Pair \\ p & \longmapsto & p+2 \end{array}$$

- $\mathbb{N} \setminus \{1\}, \mathbb{N} \setminus \{1,3\}, \mathbb{N} \setminus \{1,3,5\}, \ldots$ vérifient aussi les 3 propriétés de la définition, mais *Pair* est le plus petit d'entre eux : $Pair \subseteq \mathbb{N} \setminus \{1\}$, $Pair \subseteq \mathbb{N} \setminus \{1,3\}$, $Pair \subseteq \mathbb{N} \setminus \{1,3,5\}, \ldots$

Exemple

L'ensemble *Pair* $\subseteq \mathbb{N}$

1. *(Base)* $\{0\}$
2. *(Règle)* si $p \in Pair$ alors $p+2 \in Pair$

Remarques

- La base ne contient qu'un seul élément
- Il y a une seule règle, c'est une règle de construction interne
- que l'on pourrait définir plus formellement comme

$$\mathbf{r} : \begin{array}{rcl} Pair & \longrightarrow & Pair \\ p & \longmapsto & p+2 \end{array}$$

- $\mathbb{N} \setminus \{1\}, \mathbb{N} \setminus \{1,3\}, \mathbb{N} \setminus \{1,3,5\}, \ldots$ vérifient aussi les 3 propriétés de la définition, mais *Pair* est le plus petit d'entre eux :
  $Pair \subseteq \mathbb{N} \setminus \{1\}$, $Pair \subseteq \mathbb{N} \setminus \{1,3\}$, $Pair \subseteq \mathbb{N} \setminus \{1,3,5\}, \ldots$

Exemple

L'ensemble *Pair* $\subseteq \mathbb{N}$

1. *(Base)* $\{0\}$
2. *(Règle)* si $p \in Pair$ alors $p + 2 \in Pair$

Remarques

- La base ne contient qu'un seul élément
- Il y a une seule règle, c'est une règle de construction interne
- que l'on pourrait définir plus formellement comme

$$\begin{array}{rccc} \mathbf{r} : & Pair & \longrightarrow & Pair \\ & p & \longmapsto & p+2 \end{array}$$

- $\mathbb{N} \setminus \{1\}, \mathbb{N} \setminus \{1,3\}, \mathbb{N} \setminus \{1,3,5\}, \ldots$ vérifient aussi les 3 propriétés de la définition, mais *Pair* est le plus petit d'entre eux : $Pair \subseteq \mathbb{N} \setminus \{1\}$, $Pair \subseteq \mathbb{N} \setminus \{1,3\}$, $Pair \subseteq \mathbb{N} \setminus \{1,3,5\}, \ldots$

Exemple

L'ensemble *Pair* $\subseteq \mathbb{N}$

1. *(Base)* $\{0\}$
2. *(Règle)* si $p \in Pair$ alors $p + 2 \in Pair$

Remarques

- La base ne contient qu'un seul élément
- Il y a une seule règle, c'est une règle de construction interne
- que l'on pourrait définir plus formellement comme
$$\begin{array}{rccc} \mathbf{r} : & Pair & \longrightarrow & Pair \\ & p & \longmapsto & p+2 \end{array}$$
- $\mathbb{N} \setminus \{1\}, \mathbb{N} \setminus \{1,3\}, \mathbb{N} \setminus \{1,3,5\}, \ldots$ vérifient aussi les 3 propriétés de la définition, mais *Pair* est le plus petit d'entre eux :
$Pair \subseteq \mathbb{N} \setminus \{1\}$, $Pair \subseteq \mathbb{N} \setminus \{1,3\}$, $Pair \subseteq \mathbb{N} \setminus \{1,3,5\}, \ldots$

Exemple

L'ensemble *Pair* $\subseteq \mathbb{N}$

1. *(Base)* $\{0\}$
2. *(Règle)* si $p \in Pair$ alors $p + 2 \in Pair$

Remarques

- La base ne contient qu'un seul élément
- Il y a une seule règle, c'est une règle de construction interne
- que l'on pourrait définir plus formellement comme
$$\mathbf{r} : \begin{array}{ccc} Pair & \longrightarrow & Pair \\ p & \longmapsto & p+2 \end{array}$$
- $\mathbb{N} \setminus \{1\}, \mathbb{N} \setminus \{1,3\}, \mathbb{N} \setminus \{1,3,5\}, \ldots$ vérifient aussi les 3 propriétés de la définition, mais *Pair* est le plus petit d'entre eux :
$Pair \subseteq \mathbb{N} \setminus \{1\}$, $Pair \subseteq \mathbb{N} \setminus \{1,3\}$, $Pair \subseteq \mathbb{N} \setminus \{1,3,5\}, \ldots$

Exemple

L'ensemble *Pair* $\subseteq \mathbb{N}$

1. *(Base)* $\{0\}$
2. *(Règle)* si $p \in Pair$ alors $p + 2 \in Pair$

Remarques

- La base ne contient qu'un seul élément
- Il y a une seule règle, c'est une règle de construction interne
- que l'on pourrait définir plus formellement comme

$$\begin{array}{rccc} \mathbf{r} : & Pair & \longrightarrow & Pair \\ & p & \longmapsto & p+2 \end{array}$$

- $\mathbb{N} \setminus \{1\}, \mathbb{N} \setminus \{1,3\}, \mathbb{N} \setminus \{1,3,5\}, \ldots$ vérifient aussi les 3 propriétés de la définition, mais *Pair* est le plus petit d'entre eux :
  $Pair \subseteq \mathbb{N} \setminus \{1\}, Pair \subseteq \mathbb{N} \setminus \{1,3\}, Pair \subseteq \mathbb{N} \setminus \{1,3,5\}, \ldots$

Exemple

L'ensemble $Pair \subseteq \mathbb{N}$

1. *(Base)* $\{0\}$
2. *(Règle)* si $p \in Pair$ alors $p + 2 \in Pair$

Remarques

- La base ne contient qu'un seul élément
- Il y a une seule règle, c'est une règle de construction interne
- que l'on pourrait définir plus formellement comme

$$\mathbf{r} : \begin{array}{ccc} Pair & \longrightarrow & Pair \\ p & \longmapsto & p + 2 \end{array}$$

- $\mathbb{N} \setminus \{1\}, \mathbb{N} \setminus \{1, 3\}, \mathbb{N} \setminus \{1, 3, 5\}, \ldots$ vérifient aussi les 3 propriétés de la définition, mais $Pair$ est le plus petit d'entre eux :
$Pair \subseteq \mathbb{N} \setminus \{1\}$, $Pair \subseteq \mathbb{N} \setminus \{1, 3\}$, $Pair \subseteq \mathbb{N} \setminus \{1, 3, 5\}, \ldots$

Exemple

L'ensemble $\mathit{Pair} \subseteq \mathbb{N}$

1. *(Base)* $\{0\}$
2. *(Règle)* si $p \in \mathit{Pair}$ alors $p + 2 \in \mathit{Pair}$

Remarques

- La base ne contient qu'un seul élément
- Il y a une seule règle, c'est une règle de construction interne
- que l'on pourrait définir plus formellement comme
  $$\begin{array}{rcccc} \mathbf{r} : & \mathit{Pair} & \longrightarrow & \mathit{Pair} \\ & p & \longmapsto & p+2 \end{array}$$
- $\mathbb{N} \setminus \{1\}, \mathbb{N} \setminus \{1,3\}, \mathbb{N} \setminus \{1,3,5\}, \ldots$ vérifient aussi les 3 propriétés de la définition, mais *Pair* est le plus petit d'entre eux :
  $\mathit{Pair} \subseteq \mathbb{N} \setminus \{1\}$, $\mathit{Pair} \subseteq \mathbb{N} \setminus \{1,3\}$, $\mathit{Pair} \subseteq \mathbb{N} \setminus \{1,3,5\}, \ldots$

Exemple

L'ensemble $Pair \subseteq \mathbb{N}$

1. *(Base)* $\{0\}$
2. *(Règle)* si $p \in Pair$ alors $p + 2 \in Pair$

Remarques

- La base ne contient qu'un seul élément
- Il y a une seule règle, c'est une règle de construction interne
- que l'on pourrait définir plus formellement comme

$$\begin{array}{rccc} \mathbf{r}: & Pair & \longrightarrow & Pair \\ & p & \longmapsto & p+2 \end{array}$$

- $\mathbb{N} \setminus \{1\}, \mathbb{N} \setminus \{1,3\}, \mathbb{N} \setminus \{1,3,5\}, \ldots$ vérifient aussi les 3 propriétés de la définition, mais *Pair* est le plus petit d'entre eux : $Pair \subseteq \mathbb{N} \setminus \{1\}$, $Pair \subseteq \mathbb{N} \setminus \{1,3\}$, $Pair \subseteq \mathbb{N} \setminus \{1,3,5\}, \ldots$

Exemple

L'ensemble $\mathit{Pair} \subseteq \mathbb{N}$

1. *(Base)* $\{0\}$
2. *(Règle)* si $p \in \mathit{Pair}$ alors $p + 2 \in \mathit{Pair}$

Remarques

- La base ne contient qu'un seul élément
- Il y a une seule règle, c'est une règle de construction interne
- que l'on pourrait définir plus formellement comme

$$\mathbf{r} : \begin{array}{ccc} \mathit{Pair} & \longrightarrow & \mathit{Pair} \\ p & \longmapsto & p+2 \end{array}$$

- $\mathbb{N} \setminus \{1\}$, $\mathbb{N} \setminus \{1,3\}$, $\mathbb{N} \setminus \{1,3,5\}$, ... vérifient aussi les 3 propriétés de la définition, mais Pair est le plus petit d'entre eux :
  $\mathit{Pair} \subseteq \mathbb{N} \setminus \{1\}$, $\mathit{Pair} \subseteq \mathbb{N} \setminus \{1,3\}$, $\mathit{Pair} \subseteq \mathbb{N} \setminus \{1,3,5\}$, ...

Exemple

L'ensemble $\mathit{Pair} \subseteq \mathbb{N}$

1. *(Base)* $\{0\}$
2. *(Règle)* si $p \in \mathit{Pair}$ alors $p + 2 \in \mathit{Pair}$

Remarques

- La base ne contient qu'un seul élément
- Il y a une seule règle, c'est une règle de construction interne
- que l'on pourrait définir plus formellement comme
  $$\begin{array}{rccc} \mathbf{r}: & \mathit{Pair} & \longrightarrow & \mathit{Pair} \\ & p & \longmapsto & p+2 \end{array}$$
- $\mathbb{N} \setminus \{1\}, \mathbb{N} \setminus \{1,3\}, \mathbb{N} \setminus \{1,3,5\}, \ldots$ vérifient aussi les 3 propriétés de la définition, mais *Pair* est le plus petit d'entre eux :
  $\mathit{Pair} \subseteq \mathbb{N} \setminus \{1\}$, $\mathit{Pair} \subseteq \mathbb{N} \setminus \{1,3\}$, $\mathit{Pair} \subseteq \mathbb{N} \setminus \{1,3,5\}, \ldots$

D'autres ensembles définis par induction

- Soit A un ensemble de lettres appelé alphabet
- On définit A^* le monoïde libre sur A, l'ensemble des mots de A
- On note $u = a_1 a_2 \ldots a_n$ le mot $(a_1, a_2, \ldots, a_n)$. Sa longueur est $|u| = n$
  Le mot de longueur 0 est noté ϵ
- L'opération du monoïde est la concaténation notée . telle que
  $(a_1, \ldots, a_n).(b_1, \ldots, b_p) = (a_1, \ldots, a_n, b_1, \ldots, b_p)$.

Définition inductive du monoïde A^*

- *(Base)* $\{\epsilon\} \cup A$
- *(Règle)* $\forall u, v \in A^* : u.v \in A^*$

Exemple avec $A = \{c, o, u\}$

- La base de $\{c, o, u\}^*$ est $\{\epsilon, c, o, u\}$
- Des exemples de mots dans $\{c, o, u\}^*$: ϵ, *c*, *o*, *u*, *co*, *cou*, *coucou*
  mais aussi *ccccccc*, *coco*, *cuocuoooccuocucocuco* et bien d'autres

D'autres ensembles définis par induction

- Soit A un ensemble de lettres appelé alphabet
- On définit A^* le monoïde libre sur A, l'ensemble des mots de A
- On note $u = a_1 a_2 \ldots a_n$ le mot $(a_1, a_2, \ldots, a_n)$. Sa longueur est $|u| = n$
  Le mot de longueur 0 est noté ϵ
- L'opération du monoïde est la concaténation notée . telle que $(a_1, \ldots, a_n).(b_1, \ldots, b_p) = (a_1, \ldots, a_n, b_1, \ldots, b_p)$.

Définition inductive du monoïde A^*

- *(Base)* $\{\epsilon\} \cup A$
- *(Règle)* $\forall u, v \in A^* : u.v \in A^*$

Exemple avec $A = \{c, o, u\}$

- La base de $\{c, o, u\}^*$ est $\{\epsilon, c, o, u\}$
- Des exemples de mots dans $\{c, o, u\}^*$: ϵ, *c*, *o*, *u*, *co*, *cou*, *coucou*
  mais aussi *ccccccc*, *coco*, *cuocuooccuocucocuco* et bien d'autres

D'autres ensembles définis par induction

- Soit A un ensemble de lettres appelé alphabet
- On définit A^* le monoïde libre sur A, l'ensemble des mots de A
- On note $u = a_1 a_2 \dots a_n$ le mot $(a_1, a_2, \dots, a_n)$. Sa longueur est $|u| = n$
  Le mot de longueur 0 est noté ϵ
- L'opération du monoïde est la concaténation notée . telle que $(a_1, \dots, a_n).(b_1, \dots, b_p) = (a_1, \dots, a_n, b_1, \dots, b_p)$.

Définition inductive du monoïde A^*

- *(Base)* $\{\epsilon\} \cup A$
- *(Règle)* $\forall u, v \in A^* : u.v \in A^*$

Exemple avec $A = \{c, o, u\}$

- La base de $\{c, o, u\}^*$ est $\{\epsilon, c, o, u\}$
- Des exemples de mots dans $\{c, o, u\}^*$: ϵ, *c*, *o*, *u*, *co*, *cou*, *coucou* mais aussi *ccccccc*, *coco*, *cuocuoooccuocucocuco* et bien d'autres

D'autres ensembles définis par induction

- Soit A un ensemble de lettres appelé alphabet
- On définit A^* le monoïde libre sur A, l'ensemble des mots de A
- On note $u = a_1 a_2 \dots a_n$ le mot $(a_1, a_2, \dots, a_n)$. Sa longueur est $|u| = n$
  Le mot de longueur 0 est noté ϵ
- L'opération du monoïde est la concaténation notée . telle que $(a_1, \dots, a_n).(b_1, \dots, b_p) = (a_1, \dots, a_n, b_1, \dots, b_p)$.

Définition inductive du monoïde A^*

- *(Base)* $\{\epsilon\} \cup A$
- *(Règle)* $\forall u, v \in A^* : u.v \in A^*$

Exemple avec $A = \{c, o, u\}$

- La base de $\{c, o, u\}^*$ est $\{\epsilon, c, o, u\}$
- Des exemples de mots dans $\{c, o, u\}^*$: ϵ, *c*, *o*, *u*, *co*, *cou*, *coucou* mais aussi *ccccccc*, *coco*, *cuocuoooccuocucococuco* et bien d'autres

D'autres ensembles définis par induction

- Soit A un ensemble de lettres appelé alphabet
- On définit A^* le monoïde libre sur A, l'ensemble des mots de A
- On note $u = a_1 a_2 \dots a_n$ le mot $(a_1, a_2, \dots, a_n)$. Sa longueur est $|u| = n$
  Le mot de longueur 0 est noté ϵ
- L'opération du monoïde est la concaténation notée $.$ telle que $(a_1, \dots, a_n).(b_1, \dots, b_p) = (a_1, \dots, a_n, b_1, \dots, b_p)$.

Définition inductive du monoïde A^*

- *(Base)* $\{\epsilon\} \cup A$
- *(Règle)* $\forall u, v \in A^* : u.v \in A^*$

Exemple avec $A = \{c, o, u\}$

- La base de $\{c, o, u\}^*$ est $\{\epsilon, c, o, u\}$
- Des exemples de mots dans $\{c, o, u\}^*$: ϵ, c, o, u, *co*, *cou*, *coucou*
  mais aussi *ccccccc*, *coco*, *cuocuoooccuocucocuco* et bien d'autres

D'autres ensembles définis par induction

- Soit A un ensemble de lettres appelé alphabet
- On définit A^* le monoïde libre sur A, l'ensemble des mots de A
- On note $u = a_1 a_2 \dots a_n$ le mot $(a_1, a_2, \dots, a_n)$. Sa longueur est $|u| = n$
  Le mot de longueur 0 est noté ϵ
- L'opération du monoïde est la concaténation notée $.$ telle que $(a_1, \dots, a_n).(b_1, \dots, b_p) = (a_1, \dots, a_n, b_1, \dots, b_p)$.

Définition inductive du monoïde A^*

- *(Base)* $\{\epsilon\} \cup A$
- *(Règle)* $\forall u, v \in A^* : u.v \in A^*$

Exemple avec $A = \{c, o, u\}$

- La base de $\{c, o, u\}^*$ est $\{\epsilon, c, o, u\}$
- Des exemples de mots dans $\{c, o, u\}^*$: ϵ, *c*, *o*, *u*, *co*, *cou*, *coucou* mais aussi *ccccccc*, *coco*, *cuocuoooccuocucocuco* et bien d'autres

D'autres ensembles définis par induction

- Soit A un ensemble de lettres appelé alphabet
- On définit A^* le monoïde libre sur A, l'ensemble des mots de A
- On note $u = a_1 a_2 \dots a_n$ le mot $(a_1, a_2, \dots, a_n)$. Sa longueur est $|u| = n$
  Le mot de longueur 0 est noté ϵ
- L'opération du monoïde est la concaténation notée $.$ telle que $(a_1, \dots, a_n).(b_1, \dots, b_p) = (a_1, \dots, a_n, b_1, \dots, b_p)$.

Définition inductive du monoïde A^*

- *(Base)* $\{\epsilon\} \cup A$
- *(Règle)* $\forall u, v \in A^* : u.v \in A^*$

Exemple avec $A = \{c, o, u\}$

- La base de $\{c, o, u\}^*$ est $\{\epsilon, c, o, u\}$
- Des exemples de mots dans $\{c, o, u\}^*$: ϵ, c, o, u, *co*, *cou*, *coucou*
  mais aussi *ccccccc*, *coco*, *cuocuoooccuocucocuco* et bien d'autres

D'autres ensembles définis par induction

- Soit A un ensemble de lettres appelé alphabet
- On définit A^* le monoïde libre sur A, l'ensemble des mots de A
- On note $u = a_1 a_2 \dots a_n$ le mot $(a_1, a_2, \dots, a_n)$. Sa longueur est $|u| = n$
  Le mot de longueur 0 est noté ϵ
- L'opération du monoïde est la concaténation notée $.$ telle que $(a_1, \dots, a_n).(b_1, \dots, b_p) = (a_1, \dots, a_n, b_1, \dots, b_p)$.

Définition inductive du monoïde A^*

- *(Base)* $\{\epsilon\} \cup A$
- *(Règle)* $\forall u, v \in A^* : u.v \in A^*$

Exemple avec $A = \{c, o, u\}$

- La base de $\{c, o, u\}^*$ est $\{\epsilon, c, o, u\}$
- Des exemples de mots dans $\{c, o, u\}^*$: ϵ, *c*, *o*, *u*, *co*, *cou*, *coucou*
  mais aussi *ccccccc*, *coco*, *cuocuooocсuocucocuco* et bien d'autres

D'autres ensembles définis par induction

- Soit A un ensemble de lettres appelé alphabet
- On définit A^* le monoïde libre sur A, l'ensemble des mots de A
- On note $u = a_1 a_2 \dots a_n$ le mot $(a_1, a_2, \dots, a_n)$. Sa longueur est $|u| = n$
  Le mot de longueur 0 est noté ϵ
- L'opération du monoïde est la concaténation notée $.$ telle que $(a_1, \dots, a_n).(b_1, \dots, b_p) = (a_1, \dots, a_n, b_1, \dots, b_p)$.

Définition inductive du monoïde A^*

- *(Base)* $\{\epsilon\} \cup A$
- *(Règle)* $\forall u, v \in A^* : u.v \in A^*$

Exemple avec $A = \{c, o, u\}$

- La base de $\{c, o, u\}^*$ est $\{\epsilon, c, o, u\}$
- Des exemples de mots dans $\{c, o, u\}^*$: ϵ, *c*, *o*, *u*, *co*, *cou*, *coucou* mais aussi *ccccccc*, *coco*, *cuocuooocuocucocuco* et bien d'autres

D'autres ensembles définis par induction

- Soit A un ensemble de lettres appelé alphabet
- On définit A^* le monoïde libre sur A, l'ensemble des mots de A
- On note $u = a_1 a_2 \dots a_n$ le mot $(a_1, a_2, \dots, a_n)$. Sa longueur est $|u| = n$
  Le mot de longueur 0 est noté ϵ
- L'opération du monoïde est la concaténation notée $.$ telle que
  $(a_1, \dots, a_n).(b_1, \dots, b_p) = (a_1, \dots, a_n, b_1, \dots, b_p)$.

Définition inductive du monoïde A^*

- *(Base)* $\{\epsilon\} \cup A$
- *(Règle)* $\forall u, v \in A^* : u.v \in A^*$

Exemple avec $A = \{c, o, u\}$

- La base de $\{c, o, u\}^*$ est $\{\epsilon, c, o, u\}$
- Des exemples de mots dans $\{c, o, u\}^*$: ϵ, c, o, u, co, cou, $coucou$
  mais aussi $ccccccc$, $coco$, $cuocuooocuocucocuco$ et bien d'autres

D'autres ensembles définis par induction

- Soit A un ensemble de lettres appelé alphabet
- On définit A^* le monoïde libre sur A, l'ensemble des mots de A
- On note $u = a_1 a_2 \dots a_n$ le mot $(a_1, a_2, \dots, a_n)$. Sa longueur est $|u| = n$
  Le mot de longueur 0 est noté ϵ
- L'opération du monoïde est la concaténation notée $.$ telle que $(a_1, \dots, a_n).(b_1, \dots, b_p) = (a_1, \dots, a_n, b_1, \dots, b_p)$.

Définition inductive du monoïde A^*

- *(Base)* $\{\epsilon\} \cup A$
- *(Règle)* $\forall u, v \in A^* : u.v \in A^*$

Exemple avec $A = \{c, o, u\}$

- La base de $\{c, o, u\}^*$ est $\{\epsilon, c, o, u\}$
- Des exemples de mots dans $\{c, o, u\}^*$: ϵ, c, o, u, co, cou, $coucou$ mais aussi $ccccccc$, $coco$, $cuocuoooccuocucocuco$ et bien d'autres

D’autres ensembles définis par induction

- Soit A un ensemble de lettres appelé alphabet
- On définit A^* le monoïde libre sur A, l’ensemble des mots de A
- On note $u = a_1a_2\dots a_n$ le mot $(a_1, a_2, \dots, a_n)$. Sa longueur est $|u| = n$
  Le mot de longueur 0 est noté ϵ
- L’opération du monoïde est la concaténation notée $.$ telle que $(a_1, \dots, a_n).(b_1, \dots, b_p) = (a_1, \dots, a_n, b_1, \dots, b_p)$.

Définition inductive du monoïde A^*

- *(Base)* $\{\epsilon\} \cup A$
- *(Règle)* $\forall u, v \in A^* : u.v \in A^*$

Exemple avec $A = \{c, o, u\}$

- La base de $\{c, o, u\}^*$ est $\{\epsilon, c, o, u\}$
- Des exemples de mots dans $\{c, o, u\}^*$: ϵ, c, o, u, co, cou, $coucou$
  mais aussi $ccccccc$, $coco$, $cuocuoooccuocucococuco$ et bien d’autres

D'autres ensembles définis par induction

- Soit A un ensemble de lettres appelé alphabet
- On définit A^* le monoïde libre sur A, l'ensemble des mots de A
- On note $u = a_1 a_2 \dots a_n$ le mot $(a_1, a_2, \dots, a_n)$. Sa longueur est $|u| = n$
  Le mot de longueur 0 est noté ϵ
- L'opération du monoïde est la concaténation notée $.$ telle que $(a_1, \dots, a_n).(b_1, \dots, b_p) = (a_1, \dots, a_n, b_1, \dots, b_p)$.

Définition inductive du monoïde A^*

- *(Base)* $\{\epsilon\} \cup A$
- *(Règle)* $\forall u, v \in A^* : u.v \in A^*$

Exemple avec $A = \{c, o, u\}$

- La base de $\{c, o, u\}^*$ est $\{\epsilon, c, o, u\}$
- Des exemples de mots dans $\{c, o, u\}^*$: ϵ, *c*, *o*, *u*, *co*, *cou*, *coucou* mais aussi *ccccccc*, *coco*, *cuocuooocc uocucocuco* et bien d'autres

D'autres ensembles définis par induction

- Soit A un ensemble de lettres appelé alphabet
- On définit A^* le monoïde libre sur A, l'ensemble des mots de A
- On note $u = a_1 a_2 \dots a_n$ le mot $(a_1, a_2, \dots, a_n)$. Sa longueur est $|u| = n$
  Le mot de longueur 0 est noté ϵ
- L'opération du monoïde est la concaténation notée $.$ telle que $(a_1, \dots, a_n).(b_1, \dots, b_p) = (a_1, \dots, a_n, b_1, \dots, b_p)$.

Définition inductive du monoïde A^*

- *(Base)* $\{\epsilon\} \cup A$
- *(Règle)* $\forall u, v \in A^* : u.v \in A^*$

Exemple avec $A = \{c, o, u\}$

- La base de $\{c, o, u\}^*$ est $\{\epsilon, c, o, u\}$
- Des exemples de mots dans $\{c, o, u\}^*$: ϵ, c, o, u, co, cou, $coucou$
  mais aussi $ccccccc$, $coco$, $cuocuoooccuocucococuco$ et bien d'autres

D'autres ensembles définis par induction

- Soit A un ensemble de lettres appelé alphabet
- On définit A^* le monoïde libre sur A, l'ensemble des mots de A
- On note $u = a_1 a_2 \dots a_n$ le mot $(a_1, a_2, \dots, a_n)$. Sa longueur est $|u| = n$
  Le mot de longueur 0 est noté ϵ
- L'opération du monoïde est la concaténation notée $.$ telle que $(a_1, \dots, a_n).(b_1, \dots, b_p) = (a_1, \dots, a_n, b_1, \dots, b_p)$.

Définition inductive du monoïde A^*

- *(Base)* $\{\epsilon\} \cup A$
- *(Règle)* $\forall u, v \in A^* : u.v \in A^*$

Exemple avec $A = \{c, o, u\}$

- La base de $\{c, o, u\}^*$ est $\{\epsilon, c, o, u\}$
- Des exemples de mots dans $\{c, o, u\}^*$: ϵ, c, o, u, co, cou, $coucou$
  mais aussi $ccccccc$, $coco$, $cuocuoooccuocucocuco$ et bien d'autres

D'autres ensembles définis par induction

- Soit A un ensemble de lettres appelé alphabet
- On définit A^* le monoïde libre sur A, l'ensemble des mots de A
- On note $u = a_1 a_2 \dots a_n$ le mot $(a_1, a_2, \dots, a_n)$. Sa longueur est $|u| = n$
  Le mot de longueur 0 est noté ϵ
- L'opération du monoïde est la concaténation notée $.$ telle que $(a_1, \dots, a_n).(b_1, \dots, b_p) = (a_1, \dots, a_n, b_1, \dots, b_p)$.

Définition inductive du monoïde A^*

- *(Base)* $\{\epsilon\} \cup A$
- *(Règle)* $\forall u, v \in A^* : u.v \in A^*$

Exemple avec $A = \{c, o, u\}$

- La base de $\{c, o, u\}^*$ est $\{\epsilon, c, o, u\}$
- Des exemples de mots dans $\{c, o, u\}^*$: ϵ, *c*, *o*, *u*, *co*, *cou*, *coucou*
  mais aussi *ccccccc*, *coco*, *cuocuoooccuocucocuco* et bien d'autres

D'autres ensembles définis par induction

- Soit *Listes* l'ensemble des listes d'entiers, vu au semestre 1
- Une liste l est soit vide soit un n-uplet d'entiers
  c.-à-d. $l = \texttt{Vide}$ ou $l \in \mathbb{N}^n$ où $n \in \mathbb{N}^*$
- $\texttt{consL}$ est l'opérateur de construction des listes :

$$\texttt{consL} : \mathbb{N} \times \textit{Listes} \longrightarrow \textit{Listes}$$

  $\texttt{consL}(\texttt{p}, \texttt{L})$ ajoute l'entier p en tête de la liste L
  Ex : $\texttt{consL}(3, [1, 2]) = [3, 1, 2]$

Définition inductive de *Listes*

- *(Base)* $\{\texttt{Vide}\}$
- *(Règle)* $\forall L \in \textit{Listes}, \forall p \in \mathbb{N} : \texttt{consL}(\texttt{p}, \texttt{L}) \in \textit{Listes}$

Remarques

- Une seule règle de construction, externe
- *Listes* contient *toutes* les listes d'entiers

D'autres ensembles définis par induction

- Soit *Listes* l'ensemble des listes d'entiers, vu au semestre 1
- Une liste l est soit vide soit un n-uplet d'entiers
  c.-à-d. $l = \mathtt{Vide}$ ou $l \in \mathbb{N}^n$ où $n \in \mathbb{N}^*$
- $\mathtt{consL}$ est l'opérateur de construction des listes :
  $$\mathtt{consL} : \mathbb{N} \times \mathit{Listes} \longrightarrow \mathit{Listes}$$
  $\mathtt{consL}(\mathtt{p}, \mathtt{L})$ ajoute l'entier p en tête de la liste L
  Ex : $\mathtt{consL}(3, [1, 2]) = [3, 1, 2]$

Définition inductive de *Listes*

- *(Base)* $\{\mathtt{Vide}\}$
- *(Règle)* $\forall L \in \mathit{Listes}, \forall p \in \mathbb{N} : \mathtt{consL}(\mathtt{p}, \mathtt{L}) \in \mathit{Listes}$

Remarques

- Une seule règle de construction, externe
- *Listes* contient *toutes* les listes d'entiers

D'autres ensembles définis par induction

- Soit *Listes* l'ensemble des listes d'entiers, vu au semestre 1
- Une liste l est soit vide soit un *n*-uplet d'entiers
  c.-à-d. $l = \texttt{Vide}$ ou $l \in \mathbb{N}^n$ où $n \in \mathbb{N}^*$
- $\texttt{consL}$ est l'opérateur de construction des listes :
  $$\texttt{consL} : \mathbb{N} \times \textit{Listes} \longrightarrow \textit{Listes}$$
  $\texttt{consL}(\texttt{p}, \texttt{L})$ ajoute l'entier p en tête de la liste L
  Ex : $\texttt{consL}(3, [1, 2]) = [3, 1, 2]$

Définition inductive de *Listes*

- *(Base)* $\{\texttt{Vide}\}$
- *(Règle)* $\forall L \in \textit{Listes}, \forall p \in \mathbb{N} : \texttt{consL}(\texttt{p}, \texttt{L}) \in \textit{Listes}$

Remarques

- Une seule règle de construction, externe
- *Listes* contient *toutes* les listes d'entiers

D'autres ensembles définis par induction

- Soit *Listes* l'ensemble des listes d'entiers, vu au semestre 1
- Une liste l est soit vide soit un n-uplet d'entiers
  c.-à-d. $l = \texttt{Vide}$ ou $l \in \mathbb{N}^n$ où $n \in \mathbb{N}^*$
- $\texttt{consL}$ est l'opérateur de construction des listes :
  $$\texttt{consL} : \mathbb{N} \times \textit{Listes} \longrightarrow \textit{Listes}$$
  $\texttt{consL}(\texttt{p}, \texttt{L})$ ajoute l'entier p en tête de la liste L
  Ex : $\texttt{consL}(3, [1, 2]) = [3, 1, 2]$

Définition inductive de *Listes*

- *(Base)* $\{\texttt{Vide}\}$
- *(Règle)* $\forall L \in \textit{Listes}, \forall p \in \mathbb{N} : \texttt{consL}(\texttt{p}, \texttt{L}) \in \textit{Listes}$

Remarques

- Une seule règle de construction, externe
- *Listes* contient *toutes* les listes d'entiers

D'autres ensembles définis par induction

- Soit *Listes* l'ensemble des listes d'entiers, vu au semestre 1
- Une liste l est soit vide soit un n-uplet d'entiers
  c.-à-d. $l = \mathtt{Vide}$ ou $l \in \mathbb{N}^n$ où $n \in \mathbb{N}^*$
- $\mathtt{consL}$ est l'opérateur de construction des listes :
  $$\mathtt{consL} : \mathbb{N} \times \textit{Listes} \longrightarrow \textit{Listes}$$
  $\mathtt{consL}(\mathtt{p}, \mathtt{L})$ ajoute l'entier p en tête de la liste L
  Ex : $\mathtt{consL}(3, [1, 2]) = [3, 1, 2]$

Définition inductive de *Listes*

- *(Base)* $\{\mathtt{Vide}\}$
- *(Règle)* $\forall L \in \textit{Listes}, \forall p \in \mathbb{N} : \mathtt{consL}(\mathtt{p}, \mathtt{L}) \in \textit{Listes}$

Remarques

- Une seule règle de construction, externe
- *Listes* contient *toutes* les listes d'entiers

D'autres ensembles définis par induction

- Soit *Listes* l'ensemble des listes d'entiers, vu au semestre 1
- Une liste l est soit vide soit un n-uplet d'entiers
  c.-à-d. $l = \texttt{Vide}$ ou $l \in \mathbb{N}^n$ où $n \in \mathbb{N}^*$
- $\texttt{consL}$ est l'opérateur de construction des listes :
  $$\texttt{consL} : \mathbb{N} \times \textit{Listes} \longrightarrow \textit{Listes}$$
  $\texttt{consL}(\texttt{p}, \texttt{L})$ ajoute l'entier p en tête de la liste L
  Ex : $\texttt{consL}(3, [1, 2]) = [3, 1, 2]$

Définition inductive de *Listes*

- *(Base)* $\{\texttt{Vide}\}$
- *(Règle)* $\forall L \in \textit{Listes}, \forall p \in \mathbb{N} : \texttt{consL}(\texttt{p}, \texttt{L}) \in \textit{Listes}$

Remarques

- Une seule règle de construction, externe
- *Listes* contient *toutes* les listes d'entiers

D'autres ensembles définis par induction

- Soit *Listes* l'ensemble des listes d'entiers, vu au semestre 1
- Une liste l est soit vide soit un n-uplet d'entiers
  c.-à-d. $l = \texttt{Vide}$ ou $l \in \mathbb{N}^n$ où $n \in \mathbb{N}^*$
- $\texttt{consL}$ est l'opérateur de construction des listes :
  $$\texttt{consL} : \mathbb{N} \times \textit{Listes} \longrightarrow \textit{Listes}$$
  $\texttt{consL}(\texttt{p}, \texttt{L})$ ajoute l'entier p en tête de la liste L
  Ex : $\texttt{consL}(3, [1, 2]) = [3, 1, 2]$

Définition inductive de *Listes*

- *(Base)* $\{\texttt{Vide}\}$
- *(Règle)* $\forall L \in \textit{Listes}, \forall p \in \mathbb{N} : \texttt{consL}(\texttt{p}, \texttt{L}) \in \textit{Listes}$

Remarques

- Une seule règle de construction, externe
- *Listes* contient *toutes* les listes d'entiers

D'autres ensembles définis par induction

- Soit *Listes* l'ensemble des listes d'entiers, vu au semestre 1
- Une liste l est soit vide soit un n-uplet d'entiers
  c.-à-d. $l = \mathtt{Vide}$ ou $l \in \mathbb{N}^n$ où $n \in \mathbb{N}^*$
- $\mathtt{consL}$ est l'opérateur de construction des listes :
  $$\mathtt{consL} : \mathbb{N} \times \textit{Listes} \longrightarrow \textit{Listes}$$
  $\mathtt{consL(p, L)}$ ajoute l'entier p en tête de la liste L
  Ex : $\mathtt{consL(3, [1, 2]) = [3, 1, 2]}$

Définition inductive de *Listes*

- *(Base)* $\{\mathtt{Vide}\}$
- *(Règle)* $\forall L \in \textit{Listes}, \forall p \in \mathbb{N} : \mathtt{consL(p, L)} \in \textit{Listes}$

Remarques

- Une seule règle de construction, externe
- *Listes* contient *toutes* les listes d'entiers

D'autres ensembles définis par induction

- Soit *Listes* l'ensemble des listes d'entiers, vu au semestre 1
- Une liste l est soit vide soit un n-uplet d'entiers
  c.-à-d. $l = \mathtt{Vide}$ ou $l \in \mathbb{N}^n$ où $n \in \mathbb{N}^*$
- $\mathtt{consL}$ est l'opérateur de construction des listes :
  $$\mathtt{consL} : \mathbb{N} \times \mathit{Listes} \longrightarrow \mathit{Listes}$$
  $\mathtt{consL}(\mathtt{p}, \mathtt{L})$ ajoute l'entier p en tête de la liste L
  Ex : $\mathtt{consL}(3, [1, 2]) = [3, 1, 2]$

Définition inductive de *Listes*

- *(Base)* $\{\mathtt{Vide}\}$
- *(Règle)* $\forall L \in \mathit{Listes}, \forall p \in \mathbb{N} : \mathtt{consL}(\mathtt{p}, \mathtt{L}) \in \mathit{Listes}$

Remarques

- Une seule règle de construction, externe
- *Listes* contient *toutes* les listes d'entiers

D'autres ensembles définis par induction

- Soit *Listes* l'ensemble des listes d'entiers, vu au semestre 1
- Une liste l est soit vide soit un n-uplet d'entiers
  c.-à-d. $l = \mathtt{Vide}$ ou $l \in \mathbb{N}^n$ où $n \in \mathbb{N}^*$
- $\mathtt{consL}$ est l'opérateur de construction des listes :
  $$\mathtt{consL} : \mathbb{N} \times \mathit{Listes} \longrightarrow \mathit{Listes}$$
  $\mathtt{consL}(\mathtt{p}, \mathtt{L})$ ajoute l'entier p en tête de la liste L
  Ex : $\mathtt{consL}(\mathtt{3}, [\mathtt{1}, \mathtt{2}]) = [\mathtt{3}, \mathtt{1}, \mathtt{2}]$

Définition inductive de *Listes*

- *(Base)* $\{\mathtt{Vide}\}$
- *(Règle)* $\forall L \in \mathit{Listes}, \forall p \in \mathbb{N} : \mathtt{consL}(\mathtt{p}, \mathtt{L}) \in \mathit{Listes}$

Remarques

- Une seule règle de construction, externe
- *Listes* contient *toutes* les listes d'entiers

D'autres ensembles définis par induction

L'ensemble ArbreBin des arbres binaires étiquetés sur un alphabet $\mathcal{A}$ est défini inductivement par :

- *(Base)* $\{ABvide\}$
- *(Règle)* $\forall e \in \mathcal{A}, \forall g, d \in \text{ArbreBin} : (e, g, d) \in \text{ArbreBin}$

Exemple avec $\mathcal{A} = \{+, -, *, /\} \cup \mathbb{N}$

Soit $A = (+, (2, ABvide, ABvide), (*, (3, ABvide, ABvide), (4, ABvide, ABvide)))$ un arbre binaire qu'on représente graphiquement comme suit

D'autres ensembles définis par induction

L'ensemble ArbreBin des arbres binaires étiquetés sur un alphabet $\mathcal{A}$ est défini inductivement par :

- *(Base)* $\{ABvide\}$
- *(Règle)* $\forall e \in \mathcal{A}, \forall g, d \in \text{ArbreBin} : (e, g, d) \in \text{ArbreBin}$

Exemple avec $\mathcal{A} = \{+, -, *, /\} \cup \mathbb{N}$

Soit $A = (+, (2, ABvide, ABvide), (*, (3, ABvide, ABvide), (4, ABvide, ABvide)))$ un arbre binaire qu'on représente graphiquement comme suit

D'autres ensembles définis par induction

L'ensemble ArbreBin des arbres binaires étiquetés sur un alphabet $\mathcal{A}$ est défini inductivement par :

- *(Base)* $\{ABvide\}$
- *(Règle)* $\forall e \in \mathcal{A}, \forall g, d \in \text{ArbreBin} : (e, g, d) \in \text{ArbreBin}$

Exemple avec $A = \{+, -, *, /\} \cup \mathbb{N}$

Soit $A = (+, (2, ABvide, ABvide), (*, (3, ABvide, ABvide), (4, ABvide, ABvide)))$ un arbre binaire qu'on représente graphiquement comme suit

D'autres ensembles définis par induction

L'ensemble ArbreBin des arbres binaires étiquetés sur un alphabet $\mathcal{A}$ est défini inductivement par :

- *(Base)* $\{ABvide\}$
- *(Règle)* $\forall e \in \mathcal{A}, \forall g, d \in \text{ArbreBin} : (e, g, d) \in \text{ArbreBin}$

Exemple avec $\mathcal{A} = \{+, -, *, /\} \cup \mathbb{N}$

Soit $A = (+, (2, ABvide, ABvide), (*, (3, ABvide, ABvide), (4, ABvide, ABvide)))$ un arbre binaire qu'on représente graphiquement comme suit

D'autres ensembles définis par induction

L'ensemble ArbreBin des arbres binaires étiquetés sur un alphabet $\mathcal{A}$ est défini inductivement par :

- *(Base)* $\{\mathit{ABvide}\}$
- *(Règle)* $\forall e \in \mathcal{A}, \forall g, d \in \text{ArbreBin} : (e, g, d) \in \text{ArbreBin}$

Exemple avec $\mathcal{A} = \{+, -, *, /\} \cup \mathbb{N}$

Soit $A = (+, (2, \mathit{ABvide}, \mathit{ABvide}), (*, (3, \mathit{ABvide}, \mathit{ABvide}), (4, \mathit{ABvide}, \mathit{ABvide})))$ un arbre binaire qu'on représente graphiquement comme suit

D'autres ensembles définis par induction

L'ensemble ArbreBin des arbres binaires étiquetés sur un alphabet $\mathcal{A}$ est défini inductivement par :

- *(Base)* $\{ABvide\}$
- *(Règle)* $\forall e \in \mathcal{A}, \forall g, d \in \text{ArbreBin} : (e, g, d) \in \text{ArbreBin}$

Exemple avec $\mathcal{A} = \{+, -, *, /\} \cup \mathbb{N}$

Soit $A = (+, (2, ABvide, ABvide), (*, (3, ABvide, ABvide), (4, ABvide, ABvide)))$ un arbre binaire qu'on représente graphiquement comme suit

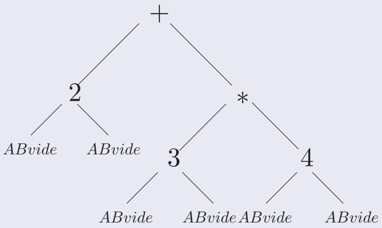
D'autres ensembles définis par induction

L'ensemble ArbreBin des arbres binaires étiquetés sur un alphabet $\mathcal{A}$ est défini inductivement par :

- *(Base)* $\{ABvide\}$
- *(Règle)* $\forall e \in \mathcal{A}, \forall g, d \in \text{ArbreBin} : (e, g, d) \in \text{ArbreBin}$

Exemple avec $\mathcal{A} = \{+, -, *, /\} \cup \mathbb{N}$

Soit $A = (+, (2, ABvide, ABvide), (*, (3, ABvide, ABvide), (4, ABvide, ABvide)))$ un arbre binaire qu'on représente graphiquement comme suit



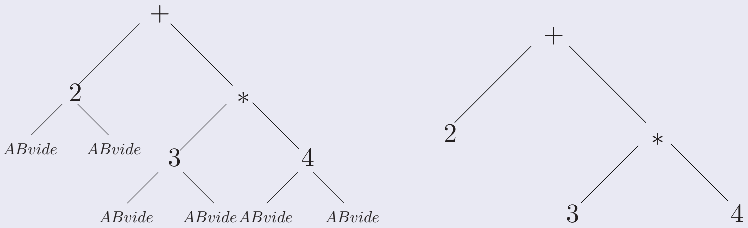
D'autres ensembles définis par induction

L'ensemble ArbreBin des arbres binaires étiquetés sur un alphabet $\mathcal{A}$ est défini inductivement par :

- *(Base)* $\{ABvide\}$
- *(Règle)* $\forall e \in \mathcal{A}, \forall g, d \in \text{ArbreBin} : (e, g, d) \in \text{ArbreBin}$

Exemple avec $\mathcal{A} = \{+, -, *, /\} \cup \mathbb{N}$

Soit $A = (+, (2, ABvide, ABvide), (*, (3, ABvide, ABvide), (4, ABvide, ABvide)))$ un arbre binaire qu'on représente graphiquement comme suit



Séquence de construction de l'exemple précédent

$\mathcal{A}$ $\{+, *, -, /, 0, 1, 2, 3, 4, 5, 6, \ldots\}$

$ArbreBin$

Séquence de construction de l'exemple précédent

$\mathcal{A}$ $\{+, *, -, /, 0, 1, 2, 3, 4, 5, 6, \ldots\}$

$ArbreBin$

$ABvide$

Séquence de construction de l'exemple précédent

$\mathcal{A}$ $\{+, *, -, /, 0, 1, 2, 3, 4, 5, 6, \ldots\}$

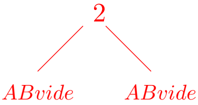
$ArbreBin$

$ABvide$

Séquence de construction de l’exemple précédent

$\mathcal{A}$ $\{+, *, -, /, 0, 1, 2, 3, 4, 5, 6, \ldots\}$

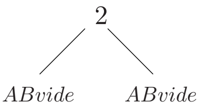
$ArbreBin$



$ABvide$

$\mathcal{A}$ $\{+, *, -, /, 0, 1, 2, 3, 4, 5, 6, \ldots\}$

$ArbreBin$

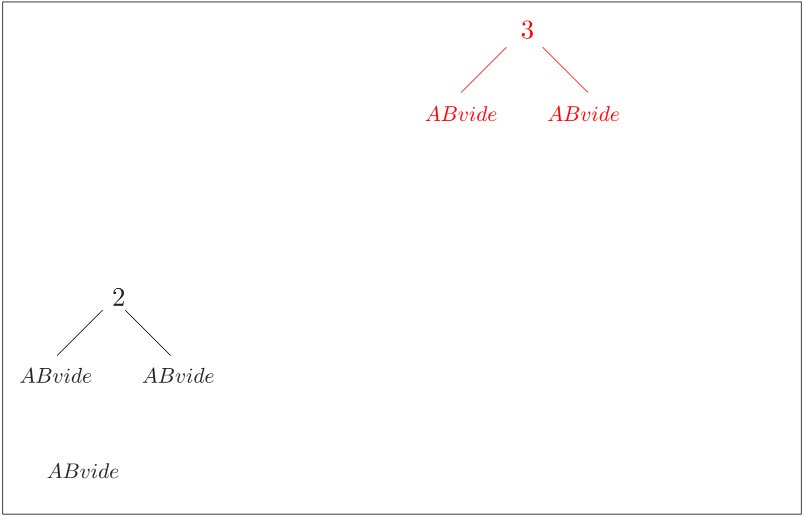


$ABvide$

Séquence de construction de l'exemple précédent

$\mathcal{A}$ $\{+, *, -, /, 0, 1, 2, 3, 4, 5, 6, \ldots\}$

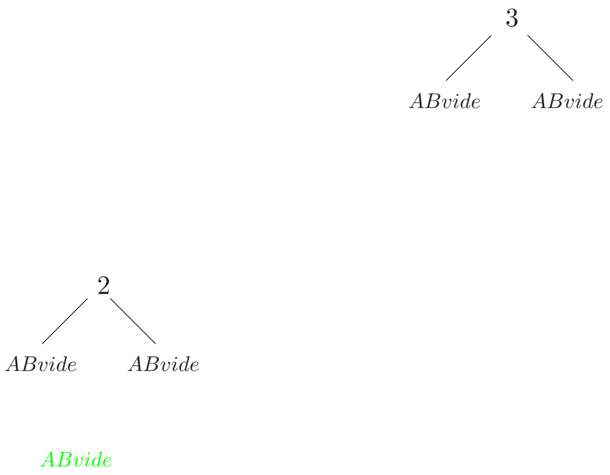
ArbreBin



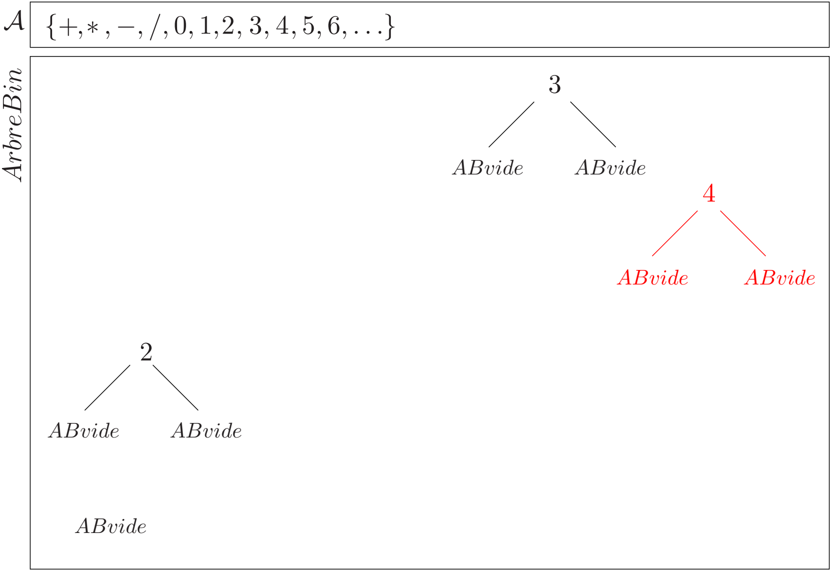
Séquence de construction de l'exemple précédent

$\mathcal{A}$ $\{+, *, -, /, 0, 1, 2, 3, 4, 5, 6, \ldots\}$

$ArbreBin$



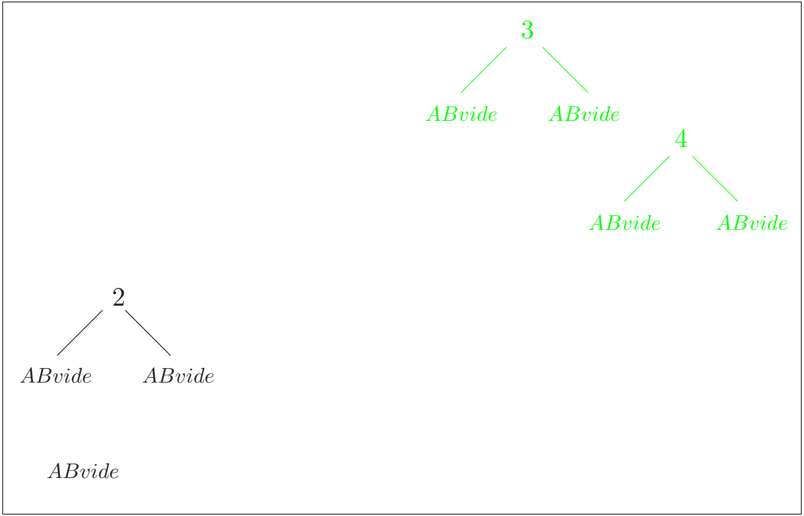
Séquence de construction de l'exemple précédent



Séquence de construction de l'exemple précédent

$\mathcal{A}$ $\{+, *, -, /, 0, 1, 2, 3, 4, 5, 6, \ldots\}$

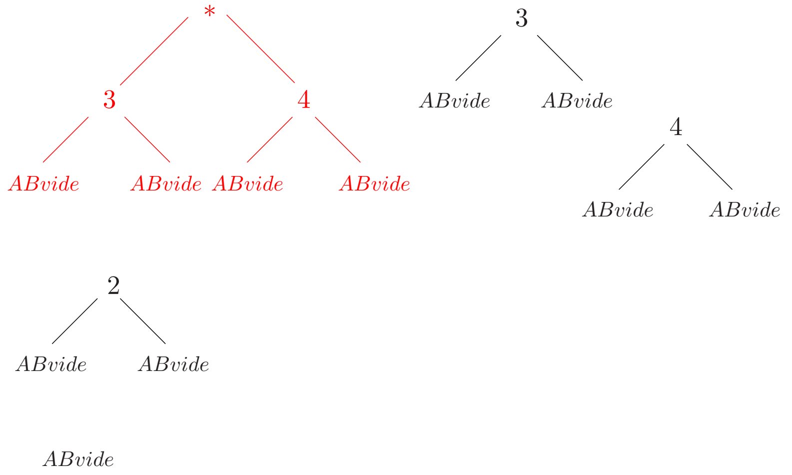
$ArbreBin$



Séquence de construction de l'exemple précédent

$\mathcal{A}$ $\{+, *, -, /, 0, 1, 2, 3, 4, 5, 6, \ldots\}$

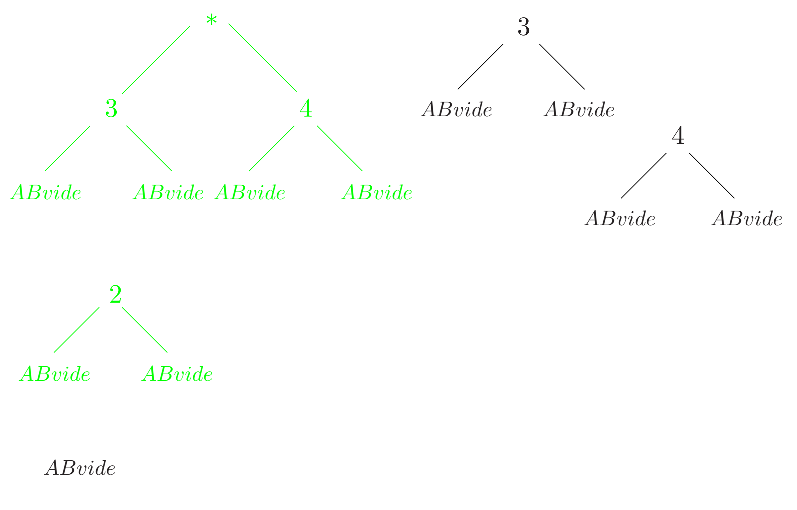
ArbreBin



Séquence de construction de l'exemple précédent

$\mathcal{A}$ $\{+, *, -, /, 0, 1, 2, 3, 4, 5, 6, \ldots\}$

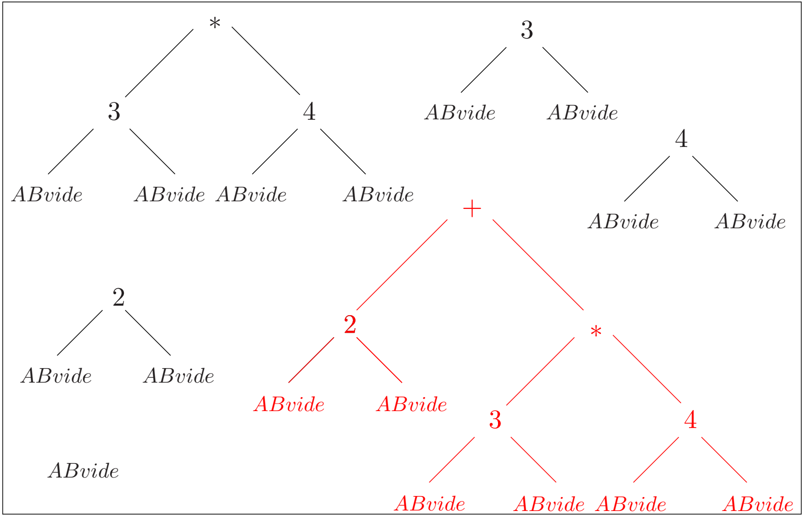
ArbreBin



Séquence de construction de l'exemple précédent

$\mathcal{A}$ $\{+, *, -, /, 0, 1, 2, 3, 4, 5, 6, \ldots\}$

$ArbreBin$



Sommaire

Fonctions définies inductivement

Définition

Soit $E \subseteq \mathcal{U}$ un ensemble défini *[de façon non ambiguë]** par induction à partir de $(\mathcal{B}, \Omega)$

On définit une fonction $\phi : E \longrightarrow F$ par induction avec :

* Intuitivement cela signifie qu'il n'existe qu'une seule façon de construire un élément $e \in E$

Fonctions définies inductivement

Définition

Soit $E \subseteq \mathcal{U}$ un ensemble défini *[de façon non ambiguë]** par induction à partir de $(\mathcal{B}, \Omega)$

On définit une fonction $\phi : E \longrightarrow F$ par induction avec :

* Intuitivement cela signifie qu'il n'existe qu'une seule façon de construire un élément $e \in E$

Fonctions définies inductivement

Définition

Soit $E \subseteq \mathcal{U}$ un ensemble défini *[de façon non ambiguë]** par induction à partir de $(\mathcal{B}, \Omega)$

On définit une fonction ϕ **:** $E \longrightarrow F$ par induction avec :

* Intuitivement cela signifie qu'il n'existe qu'une seule façon de construire un élément $e \in E$

Fonctions définies inductivement

Définition

Soit $E \subseteq \mathcal{U}$ un ensemble défini *[de façon non ambiguë]** par induction à partir de $(\mathcal{B}, \Omega)$

On définit une fonction $\phi : E \longrightarrow F$ par induction avec :

- la valeur de $\phi(x) \in F$ pour chaque $x \in \mathcal{B}$
- pour chaque règle f d'arité n de Ω, $\phi(f(x_1, \ldots, x_n))$ est une valeur qu'on exprime en fonction de $x_1, \ldots, x_n, \phi(x_1), \ldots, \phi(x_n)$

* Intuitivement cela signifie qu'il n'existe qu'une seule façon de construire un élément $e \in E$

Fonctions définies inductivement

Définition

Soit $E \subseteq \mathcal{U}$ un ensemble défini *[de façon non ambiguë]** par induction à partir de $(\mathcal{B}, \Omega)$

On définit une fonction $\phi : E \longrightarrow F$ par induction avec :

- la valeur de $\phi(x) \in F$ pour chaque $x \in \mathcal{B}$
- pour chaque règle f d'arité n de Ω, $\phi(f(x_1, \ldots, x_n))$ est une valeur qu'on exprime en fonction de $x_1, \ldots, x_n, \phi(x_1), \ldots, \phi(x_n)$

* Intuitivement cela signifie qu'il n'existe qu'une seule façon de construire un élément $e \in E$

Exemples : 3 fonctions définies par induction

La longueur d'un mot de A^*

- $|\epsilon| = 0$; $a \in A$: $|a| = 1$
- $\forall m_1, m_2 \in A^*$: $|m_1.m_2| = |m_1| + |m_2|$

La hauteur d'un arbre binaire étiqueté sur $\mathcal{A}$

- $h(ABvide) = 0$
- $e \in \mathcal{A}, g, d \in$ ArbreBin : $h((e, g, d)) = 1 + max(h(g), h(d))$

Son nombre de sous-arbres non vides d'un arbre

- $nbssab(ABvide) = 0$
- $e \in \mathcal{A}, g, d \in$ ArbreBin : $nbssab((e, g, d)) = 1 + nbssab(g) + nbssab(d)$

Rmq : un arbre est sous-arbre de lui-même

Exemples : 3 fonctions définies par induction

La longueur d'un mot de A^*

- $|\epsilon| = 0$; $a \in A$: $|a| = 1$
- $\forall m_1, m_2 \in A^*$: $|m_1.m_2| = |m_1| + |m_2|$

La hauteur d'un arbre binaire étiqueté sur $\mathcal{A}$

- $h(ABvide) = 0$
- $e \in \mathcal{A}, g, d \in$ ArbreBin : $h((e, g, d)) = 1 + max(h(g), h(d))$

Son nombre de sous-arbres non vides d'un arbre

- $nbssab(ABvide) = 0$
- $e \in \mathcal{A}, g, d \in$ ArbreBin : $nbssab((e, g, d)) = 1 + nbssab(g) + nbssab(d)$

Rmq : un arbre est sous-arbre de lui-même

Exemples : 3 fonctions définies par induction

La longueur d'un mot de A^*

- $|\epsilon| = 0$; $a \in A : |a| = 1$
- $\forall m_1, m_2 \in A^* : |m_1.m_2| = |m_1| + |m_2|$

La hauteur d'un arbre binaire étiqueté sur $\mathcal{A}$

- $h(ABvide) = 0$
- $e \in \mathcal{A}, g, d \in$ ArbreBin : $h((e, g, d)) = 1 + max(h(g), h(d))$

Son nombre de sous-arbres non vides d'un arbre

- $nbssab(ABvide) = 0$
- $e \in \mathcal{A}, g, d \in$ ArbreBin : $nbssab((e, g, d)) = 1 + nbssab(g) + nbssab(d)$

Rmq : un arbre est sous-arbre de lui-même

Exemples : 3 fonctions définies par induction

La longueur d'un mot de A^*

- $|\epsilon| = 0$; $a \in A : |a| = 1$
- $\forall m_1, m_2 \in A^* : |m_1.m_2| = |m_1| + |m_2|$

La hauteur d'un arbre binaire étiqueté sur $\mathcal{A}$

- $h(ABvide) = 0$
- $e \in \mathcal{A}, g, d \in$ ArbreBin : $h((e, g, d)) = 1 + max(h(g), h(d))$

Son nombre de sous-arbres non vides d'un arbre

- $nbssab(ABvide) = 0$
- $e \in \mathcal{A}, g, d \in$ ArbreBin : $nbssab((e, g, d)) = 1 + nbssab(g) + nbssab(d)$

Rmq : un arbre est sous-arbre de lui-même

Exemples : 3 fonctions définies par induction

La longueur d'un mot de A^*

- $|\epsilon| = 0$; $a \in A$: $|a| = 1$
- $\forall m_1, m_2 \in A^*$: $|m_1.m_2| = |m_1| + |m_2|$

La hauteur d'un arbre binaire étiqueté sur $\mathcal{A}$

- $h(ABvide) = 0$
- $e \in \mathcal{A}, g, d \in$ ArbreBin : $h((e, g, d)) = 1 + max(h(g), h(d))$

Son nombre de sous-arbres non vides d'un arbre

- $nbssab(ABvide) = 0$
- $e \in \mathcal{A}, g, d \in$ ArbreBin : $nbssab((e, g, d)) = 1 + nbssab(g) + nbssab(d)$

Rmq : un arbre est sous-arbre de lui-même

Exemples : 3 fonctions définies par induction

La longueur d'un mot de A^*

- $|\epsilon| = 0$; $a \in A : |a| = 1$
- $\forall m_1, m_2 \in A^* : |m_1.m_2| = |m_1| + |m_2|$

La hauteur d'un arbre binaire étiqueté sur $\mathcal{A}$

- $h(ABvide) = 0$
- $e \in \mathcal{A}, g, d \in \text{ArbreBin} : h((e, g, d)) = 1 + max(h(g), h(d))$

Son nombre de sous-arbres non vides d'un arbre

- $nbssab(ABvide) = 0$
- $e \in \mathcal{A}, g, d \in \text{ArbreBin} : nbssab((e, g, d)) = 1 + nbssab(g) + nbssab(d)$

Rmq : un arbre est sous-arbre de lui-même

Exemples : 3 fonctions définies par induction

La longueur d'un mot de A^*

- $|\epsilon| = 0$; $a \in A$: $|a| = 1$
- $\forall m_1, m_2 \in A^* : |m_1.m_2| = |m_1| + |m_2|$

La hauteur d'un arbre binaire étiqueté sur $\mathcal{A}$

- $h(ABvide) = 0$
- $e \in \mathcal{A}, g, d \in$ ArbreBin : $h((e, g, d)) = 1 + max(h(g), h(d))$

Son nombre de sous-arbres non vides d'un arbre

- $nbssab(ABvide) = 0$
- $e \in \mathcal{A}, g, d \in$ ArbreBin : $nbssab((e, g, d)) = 1 + nbssab(g) + nbssab(d)$

Rmq : un arbre est sous-arbre de lui-même

Sommaire

Les preuves par induction structurelle généralisent le principe de preuve par récurrence sur les entiers

$\mathbb{N}$ est d'ailleurs un ensemble défini par induction :

- *(Base)* $\{0\}$
- *(Règle)* si $n \in \mathbb{N}$ alors $n + 1 \in \mathbb{N}$

Les preuves par induction structurelle se calquent sur la définition par induction de l'ensemble, d'où "preuve par induction structurelle"

Les preuves par induction structurelle généralisent le principe de preuve par récurrence sur les entiers

$\mathbb{N}$ est d'ailleurs un ensemble défini par induction :

- *(Base)* $\{0\}$
- *(Règle)* si $n \in \mathbb{N}$ alors $n + 1 \in \mathbb{N}$

Les preuves par induction structurelle se calquent sur la définition par induction de l'ensemble, d'où "preuve par induction structurelle"

Introduction

Les preuves par induction structurelle généralisent le principe de preuve par récurrence sur les entiers

$\mathbb{N}$ est d'ailleurs un ensemble défini par induction :

- *(Base)* $\{0\}$
- *(Règle)* si $n \in \mathbb{N}$ alors $n + 1 \in \mathbb{N}$

Les preuves par induction structurelle se calquent sur la définition par induction de l'ensemble, d'où "preuve par induction structurelle"

Introduction

Les preuves par induction structurelle généralisent le principe de preuve par récurrence sur les entiers

$\mathbb{N}$ est d'ailleurs un ensemble défini par induction :

- *(Base)* $\{0\}$
- *(Règle)* si $n \in \mathbb{N}$ alors $n + 1 \in \mathbb{N}$

Les preuves par induction structurelle se calquent sur la définition par induction de l'ensemble, d'où "preuve par induction structurelle"

Preuve par induction structurelle

Soit E un ensemble défini par induction à partir de $(\mathcal{B}, \Omega)$
Soit $P\ : E \longrightarrow$ Booléen la propriété que l'on veut démontrer

Pour montrer que $P(x)$ est vraie pour chaque élément de E, il suffit de prouver

1. $\forall x \in \mathcal{B} : P(x)$ est vraie
2. Pour chaque règle interne $f_i \in \Omega$ d'arité n définie de E^n dans E et pour chaque $x_1, \ldots, x_n \in E$:
   si $P(x_1), \ldots, P(x_n)$ sont vraies alors $P(f_i(x_1, \ldots, x_n))$ est vraie.
3. Pour chaque règle externe $f_e \in \Omega$ d'arité n définie de $K_1 \times \ldots \times K_p \times E \times \ldots \times E$ dans E ($p < n$), pour chaque $(x_1, \ldots, x_p)$ dans $K_1 \times \ldots \times K_p$ et pour chaque $(x_{p+1}, \ldots, x_n)$ dans E^{n-p} :
   si $P(x_{p+1}), \ldots, P(x_n)$ sont vraies, alors $P(f_e(x_1, \ldots, x_p, x_{p+1}, \ldots, x_n))$ est vraie.

Autrement dit, <u>si</u> P est vraie pour chaque élément de la base <u>et si</u> P est conservée par l'application des règles, <u>alors</u> P est vraie pour tous les éléments de l'ensemble

Preuve par induction structurelle

Soit E un ensemble défini par induction à partir de $(\mathcal{B}, \Omega)$
Soit $P : E \longrightarrow$ Booléen la propriété que l'on veut démontrer

Pour montrer que $P(x)$ est vraie pour chaque élément de E, il suffit de prouver

1. $\forall x \in \mathcal{B} : P(x)$ est vraie
2. Pour chaque règle interne $f_i \in \Omega$ d'arité n définie de E^n dans E et pour chaque $x_1, \ldots, x_n \in E$:
   si $P(x_1), \ldots, P(x_n)$ sont vraies alors $P(f_i(x_1, \ldots, x_n))$ est vraie.
3. Pour chaque règle externe $f_e \in \Omega$ d'arité n définie de $K_1 \times \ldots \times K_p \times E \times \ldots \times E$ dans E ($p < n$), pour chaque $(x_1, \ldots, x_p)$ dans $K_1 \times \ldots \times K_p$ et pour chaque $(x_{p+1}, \ldots, x_n)$ dans E^{n-p} :
   si $P(x_{p+1}), \ldots, P(x_n)$ sont vraies, alors $P(f_e(x_1, \ldots, x_p, x_{p+1}, \ldots, x_n))$ est vraie.

Autrement dit, si P est vraie pour chaque élément de la base et si P est conservée par l'application des règles, alors P est vraie pour tous les éléments de l'ensemble

Preuve par induction structurelle

Soit E un ensemble défini par induction à partir de $(\mathcal{B}, \Omega)$
Soit $P \ : E \longrightarrow$ Booléen la propriété que l'on veut démontrer

Pour montrer que $P(x)$ est vraie pour chaque élément de E, il suffit de prouver

1. $\forall x \in \mathcal{B} : P(x)$ est vraie
2. Pour chaque règle interne $f_i \in \Omega$ d'arité n définie de E^n dans E et pour chaque $x_1, \ldots, x_n \in E$:
   si $P(x_1), \ldots, P(x_n)$ sont vraies alors $P(f_i(x_1, \ldots, x_n))$ est vraie.
3. Pour chaque règle externe $f_e \in \Omega$ d'arité n définie de $K_1 \times \ldots \times K_p \times E \times \ldots \times E$ dans E ($p < n$), pour chaque $(x_1, \ldots, x_p)$ dans $K_1 \times \ldots \times K_p$ et pour chaque $(x_{p+1}, \ldots, x_n)$ dans E^{n-p} :
   si $P(x_{p+1}), \ldots, P(x_n)$ sont vraies, alors $P(f_e(x_1, \ldots, x_p, x_{p+1}, \ldots, x_n))$ est vraie.

Autrement dit, si P est vraie pour chaque élément de la base et si P est conservée par l'application des règles, alors P est vraie pour tous les éléments de l'ensemble

Preuve par induction structurelle

Soit E un ensemble défini par induction à partir de $(\mathcal{B}, \Omega)$
Soit $P \ : E \longrightarrow$ Booléen la propriété que l'on veut démontrer

Pour montrer que $P(x)$ est vraie pour chaque élément de E, il suffit de prouver

1. $\forall x \in \mathcal{B} : P(x)$ est vraie
2. Pour chaque règle interne $f_i \in \Omega$ d'arité n définie de E^n dans E et pour chaque $x_1, \ldots, x_n \in E$:
   si $P(x_1), \ldots, P(x_n)$ sont vraies alors $P(f_i(x_1, \ldots, x_n))$ est vraie.
3. Pour chaque règle externe $f_e \in \Omega$ d'arité n définie de $K_1 \times \ldots \times K_p \times E \times \ldots \times E$ dans E ($p < n$), pour chaque $(x_1, \ldots, x_p)$ dans $K_1 \times \ldots \times K_p$ et pour chaque $(x_{p+1}, \ldots, x_n)$ dans E^{n-p} :
   si $P(x_{p+1}), \ldots, P(x_n)$ sont vraies, alors $P(f_e(x_1, \ldots, x_p, x_{p+1}, \ldots, x_n))$ est vraie.

Autrement dit, si P est vraie pour chaque élément de la base et si P est conservée par l'application des règles, alors P est vraie pour tous les éléments de l'ensemble

Preuve par induction structurelle

Soit E un ensemble défini par induction à partir de $(\mathcal{B}, \Omega)$
Soit $P \ : E \longrightarrow$ Booléen la propriété que l'on veut démontrer

Pour montrer que $P(x)$ est vraie pour chaque élément de E, il suffit de prouver

1. $\forall x \in \mathcal{B} : P(x)$ est vraie
2. Pour chaque règle interne $f_i \in \Omega$ d'arité n définie de E^n dans E et pour chaque $x_1, \ldots, x_n \in E$:
   si $P(x_1), \ldots, P(x_n)$ sont vraies alors $P(f_i(x_1, \ldots, x_n))$ est vraie.
3. Pour chaque règle externe $f_e \in \Omega$ d'arité n définie de $K_1 \times \ldots \times K_p \times E \times \ldots \times E$ dans E ($p < n$), pour chaque $(x_1, \ldots, x_p)$ dans $K_1 \times \ldots \times K_p$ et pour chaque $(x_{p+1}, \ldots, x_n)$ dans E^{n-p} :
   si $P(x_{p+1}), \ldots, P(x_n)$ sont vraies, alors $P(f_e(x_1, \ldots, x_p, x_{p+1}, \ldots, x_n))$ est vraie.

Autrement dit, si P est vraie pour chaque élément de la base et si P est conservée par l'application des règles, alors P est vraie pour tous les éléments de l'ensemble

Preuve par induction structurelle

Soit E un ensemble défini par induction à partir de $(\mathcal{B}, \Omega)$
Soit $P\ : E \longrightarrow$ Booléen la propriété que l'on veut démontrer

Pour montrer que $P(x)$ est vraie pour chaque élément de E, il suffit de prouver

1. $\forall x \in \mathcal{B} : P(x)$ est vraie
2. Pour chaque règle interne $f_i \in \Omega$ d'arité n définie de E^n dans E et pour chaque $x_1, \ldots, x_n \in E$:
   si $P(x_1), \ldots, P(x_n)$ sont vraies alors $P(f_i(x_1, \ldots, x_n))$ est vraie.
3. Pour chaque règle externe $f_e \in \Omega$ d'arité n définie de $K_1 \times \ldots \times K_p \times E \times \ldots \times E$ dans E ($p < n$), pour chaque $(x_1, \ldots, x_p)$ dans $K_1 \times \ldots \times K_p$ et pour chaque $(x_{p+1}, \ldots, x_n)$ dans E^{n-p} :
   si $P(x_{p+1}), \ldots, P(x_n)$ sont vraies, alors $P(f_e(x_1, \ldots, x_p, x_{p+1}, \ldots, x_n))$ est vraie.

Autrement dit, si P est vraie pour chaque élément de la base et si P est conservée par l'application des règles, alors P est vraie pour tous les éléments de l'ensemble

Preuve par induction structurelle

Soit E un ensemble défini par induction à partir de $(\mathcal{B}, \Omega)$
Soit $P : E \longrightarrow$ Booléen la propriété que l'on veut démontrer

Pour montrer que $P(x)$ est vraie pour chaque élément de E, il suffit de prouver

1. $\forall x \in \mathcal{B} : P(x)$ est vraie
2. Pour chaque règle interne $f_i \in \Omega$ d'arité n définie de E^n dans E et pour chaque $x_1, \ldots, x_n \in E$:
   si $P(x_1), \ldots, P(x_n)$ sont vraies alors $P(f_i(x_1, \ldots, x_n))$ est vraie.
3. Pour chaque règle externe $f_e \in \Omega$ d'arité n définie de $K_1 \times \ldots \times K_p \times E \times \ldots \times E$ dans E ($p < n$), pour chaque $(x_1, \ldots, x_p)$ dans $K_1 \times \ldots \times K_p$ et pour chaque $(x_{p+1}, \ldots, x_n)$ dans E^{n-p} :
   si $P(x_{p+1}), \ldots, P(x_n)$ sont vraies, alors $P(f_e(x_1, \ldots, x_p, x_{p+1}, \ldots, x_n))$ est vraie.

Autrement dit, si P est vraie pour chaque élément de la base et si P est conservée par l'application des règles, alors P est vraie pour tous les éléments de l'ensemble

Preuve par induction structurelle

Soit E un ensemble défini par induction à partir de $(\mathcal{B}, \Omega)$
Soit $P\ : E \longrightarrow$ Booléen la propriété que l'on veut démontrer

Pour montrer que $P(x)$ est vraie pour chaque élément de E, il suffit de prouver

1. $\forall x \in \mathcal{B} : P(x)$ est vraie
2. Pour chaque règle interne $f_i \in \Omega$ d'arité n définie de E^n dans E et pour chaque $x_1, \ldots, x_n \in E$:
   si $P(x_1), \ldots, P(x_n)$ sont vraies alors $P(f_i(x_1, \ldots, x_n))$ est vraie.
3. Pour chaque règle externe $f_e \in \Omega$ d'arité n définie de $K_1 \times \ldots \times K_p \times E \times \ldots \times E$ dans E ($p < n$), pour chaque $(x_1, \ldots, x_p)$ dans $K_1 \times \ldots \times K_p$ et pour chaque $(x_{p+1}, \ldots, x_n)$ dans E^{n-p} :
   si $P(x_{p+1}), \ldots, P(x_n)$ sont vraies, alors $P(f_e(x_1, \ldots, x_p, x_{p+1}, \ldots, x_n))$ est vraie.

Autrement dit, <u>si</u> P est vraie pour chaque élément de la base <u>et si</u> P est conservée par l'application des règles, <u>alors</u> P est vraie pour tous les éléments de l'ensemble

Exemple

Problème

Soit $AB \in$ ArbreBin

Soient

- $h = hauteur(AB)$, la lg d'un plus long chemin de la racine à une feuille
- $n = nbssab(AB)$, le nombre de sous–arbres non vides de AB

Prouver que $h \leqslant n$

Rappel

L'ensemble ArbreBin des arbres binaires étiquetés sur un alphabet $\mathcal{A}$ est défini inductivement par :

- *(Base)* $\{ABvide\}$
- *(Règle)* $\forall e \in \mathcal{A}, \forall g, d \in$ ArbreBin : $(e, g, d) \in$ ArbreBin

Exemple

Problème

Soit $AB \in \text{ArbreBin}$

Soient

- $h = hauteur(AB)$, la lg d'un plus long chemin de la racine à une feuille
- $n = nbssab(AB)$, le nombre de sous–arbres non vides de AB

Prouver que $h \leqslant n$

Rappel

L'ensemble ArbreBin des arbres binaires étiquetés sur un alphabet $\mathcal{A}$ est défini inductivement par :

- *(Base)* $\{ABvide\}$
- *(Règle)* $\forall e \in \mathcal{A}, \forall g, d \in \text{ArbreBin} : (e, g, d) \in \text{ArbreBin}$

Exemple

Problème

Soit $AB \in$ ArbreBin

Soient

- $h = hauteur(AB)$, la lg d'un plus long chemin de la racine à une feuille
- $n = nbssab(AB)$, le nombre de sous–arbres non vides de AB

Prouver que $h \leqslant n$

Rappel

L'ensemble ArbreBin des arbres binaires étiquetés sur un alphabet $\mathcal{A}$ est défini inductivement par :

- *(Base)* $\{ABvide\}$
- *(Règle)* $\forall e \in \mathcal{A}, \forall g, d \in$ ArbreBin : $(e, g, d) \in$ ArbreBin

Exemple

Problème

Soit $AB \in \text{ArbreBin}$

Soient

- $h = hauteur(AB)$, la lg d'un plus long chemin de la racine à une feuille
- $n = nbssab(AB)$, le nombre de sous–arbres non vides de AB

Prouver que $h \leqslant n$

Rappel

L'ensemble ArbreBin des arbres binaires étiquetés sur un alphabet $\mathcal{A}$ est défini inductivement par :

- *(Base)* $\{ABvide\}$
- *(Règle)* $\forall e \in \mathcal{A}, \forall g, d \in \text{ArbreBin} : (e, g, d) \in \text{ArbreBin}$

Exemple

Problème

Soit $AB \in \text{ArbreBin}$

Soient

- $h = hauteur(AB)$, la lg d'un plus long chemin de la racine à une feuille
- $n = nbssab(AB)$, le nombre de sous–arbres non vides de AB

Prouver que $h \leqslant n$

Rappel

L'ensemble ArbreBin des arbres binaires étiquetés sur un alphabet $\mathcal{A}$ est défini inductivement par :

- *(Base)* $\{ABvide\}$
- *(Règle)* $\forall e \in \mathcal{A}, \forall g, d \in \text{ArbreBin} : (e, g, d) \in \text{ArbreBin}$

Exemple

Problème

Soit $AB \in$ ArbreBin

Soient

- $h = hauteur(AB)$, la lg d'un plus long chemin de la racine à une feuille
- $n = nbssab(AB)$, le nombre de sous–arbres non vides de AB

Prouver que $h \leqslant n$

Rappel

L'ensemble ArbreBin des arbres binaires étiquetés sur un alphabet $\mathcal{A}$ est défini inductivement par :

- *(Base)* $\{ABvide\}$
- *(Règle)* $\forall e \in \mathcal{A}, \forall g, d \in$ ArbreBin : $(e, g, d) \in$ ArbreBin

Démonstration

Soit $P(a) =$ "$hauteur(a) \leqslant nbssab(a)$"

1. **Base** Montrons que le propriété est vraie pour l'élément de la base : $hauteur(ABvide) = 0$ et $nbssab(ABvide) = 0$, on a bien $hauteur(ABvide) \leqslant nbssab(ABvide)$. $P(ABvide)$ est donc vraie

2. **Règle** Soient les arbres g et $d \in ArbreBin$ de hauteurs h_1, h_2 avec un nombre de sous–arbres n_1, n_2.
Supposons que $P(g) = P(d) =$ vrai, montrons que $P(A)$ vraie quelque soit $A = (e, g, d)$:
   - pour tout arbre $A = (e, g, d)$, sa hauteur est $h = max(h_1, h_2) + 1$, et son nombre de sous-arbres non vides est $n = n_1 + n_2 + 1$
   - et comme $n_1 \geqslant h_1$ et $n_2 \geqslant h_2$
   - on a bien $n \geqslant h_1 + h_2 + 1 \geqslant max(h_1, h_2) + 1 = h$ c.-à-d. $P(A)$ vraie

Par le principe d'induction structurelle, on a $P(A)$ vraie, $\forall A \in ArbreBin$

Démonstration

Soit $P(a) =$ "$hauteur(a) \leqslant nbssab(a)$"

1. **Base** Montrons que le propriété est vraie pour l'élément de la base :
   $hauteur(ABvide) = 0$ et $nbssab(ABvide) = 0$, on a bien
   $hauteur(ABvide) \leqslant nbssab(ABvide)$. $P(ABvide)$ est donc vraie

2. **Règle** Soient les arbres g et $d \in ArbreBin$ de hauteurs h_1,h_2 avec un nombre de sous–arbres n_1, n_2.
   Supposons que $P(g) = P(d) =$ vrai, montrons que $P(A)$ vraie quelque soit $A = (e, g, d)$:
   - pour tout arbre $A = (e, g, d)$, sa hauteur est $h = max(h_1, h_2) + 1$,
     et son nombre de sous-arbres non vides est $n = n_1 + n_2 + 1$
   - et comme $n_1 \geqslant h_1$ et $n_2 \geqslant h_2$
   - on a bien $n \geqslant h_1 + h_2 + 1 \geqslant max(h_1, h_2) + 1 = h$ c.-à-d. $P(A)$ vraie

Par le principe d'induction structurelle, on a $P(A)$ vraie, $\forall A \in ArbreBin$

Démonstration

Soit $P(a) =$ "$hauteur(a) \leqslant nbssab(a)$"

1. **Base** Montrons que le propriété est vraie pour l'élément de la base : $hauteur(ABvide) = 0$ et $nbssab(ABvide) = 0$, on a bien $hauteur(ABvide) \leqslant nbssab(ABvide)$. $P(ABvide)$ est donc vraie

2. **Règle** Soient les arbres g et $d \in ArbreBin$ de hauteurs h_1,h_2 avec un nombre de sous–arbres n_1, n_2.
   Supposons que $P(g) = P(d) =$ vrai, montrons que $P(A)$ vraie quelque soit $A = (e, g, d)$:
   - pour tout arbre $A = (e, g, d)$, sa hauteur est $h = max(h_1, h_2) + 1$, et son nombre de sous-arbres non vides est $n = n_1 + n_2 + 1$
   - et comme $n_1 \geqslant h_1$ et $n_2 \geqslant h_2$
   - on a bien $n \geqslant h_1 + h_2 + 1 \geqslant max(h_1, h_2) + 1 = h$ c.-à-d. $P(A)$ vraie

Par le principe d'induction structurelle, on a $P(A)$ vraie, $\forall A \in ArbreBin$

Démonstration

Soit $P(a) =$ "$hauteur(a) \leqslant nbssab(a)$"

1. **Base** Montrons que le propriété est vraie pour l'élément de la base : $hauteur(ABvide) = 0$ et $nbssab(ABvide) = 0$, on a bien $hauteur(ABvide) \leqslant nbssab(ABvide)$. $P(ABvide)$ est donc vraie

2. **Règle** Soient les arbres g et $d \in ArbreBin$ de hauteurs h_1,h_2 avec un nombre de sous–arbres n_1, n_2.
   Supposons que $P(g) = P(d) =$ vrai, montrons que $P(A)$ vraie quelque soit $A = (e, g, d)$:
   - pour tout arbre $A = (e, g, d)$, sa hauteur est $h = max(h_1, h_2) + 1$, et son nombre de sous-arbres non vides est $n = n_1 + n_2 + 1$
   - et comme $n_1 \geqslant h_1$ et $n_2 \geqslant h_2$
   - on a bien $n \geqslant h_1 + h_2 + 1 \geqslant max(h_1, h_2) + 1 = h$ c.-à-d. $P(A)$ vraie

Par le principe d'induction structurelle, on a $P(A)$ vraie, $\forall A \in ArbreBin$

Démonstration

Soit $P(a) =$ "$hauteur(a) \leqslant nbssab(a)$"

1. **Base** Montrons que le propriété est vraie pour l'élément de la base :
   $hauteur(ABvide) =$ 0 et $nbssab(ABvide) =$ 0, on a bien
   $hauteur(ABvide) \leqslant nbssab(ABvide)$. $P(ABvide)$ est donc vraie

2. **Règle** Soient les arbres g et $d \in ArbreBin$ de hauteurs h_1,h_2 avec un nombre de sous–arbres n_1, n_2.
   Supposons que $P(g) = P(d) =$ vrai, montrons que $P(A)$ vraie quelque soit $A = (e, g, d)$:
   - pour tout arbre $A = (e, g, d)$, sa hauteur est $h = max(h_1, h_2) + 1$,
     et son nombre de sous-arbres non vides est $n = n_1 + n_2 + 1$
   - et comme $n_1 \geqslant h_1$ et $n_2 \geqslant h_2$
   - on a bien $n \geqslant h_1 + h_2 + 1 \geqslant max(h_1, h_2) + 1 = h$ c.-à-d. $P(A)$ vraie

Par le principe d'induction structurelle, on a $P(A)$ vraie, $\forall A \in ArbreBin$

Démonstration

Soit $P(a) =$ "$hauteur(a) \leqslant nbssab(a)$"

1. **Base** Montrons que le propriété est vraie pour l'élément de la base : $hauteur(ABvide) = 0$ et $nbssab(ABvide) = 0$, on a bien $hauteur(ABvide) \leqslant nbssab(ABvide)$. $P(ABvide)$ est donc vraie

2. **Règle** Soient les arbres g et $d \in ArbreBin$ de hauteurs h_1, h_2 avec un nombre de sous–arbres n_1, n_2.
   Supposons que $P(g) = P(d) =$ vrai, montrons que $P(A)$ vraie quelque soit $A = (e, g, d)$:
   - pour tout arbre $A = (e, g, d)$, sa hauteur est $h = max(h_1, h_2) + 1$, et son nombre de sous-arbres non vides est $n = n_1 + n_2 + 1$
   - et comme $n_1 \geqslant h_1$ et $n_2 \geqslant h_2$
   - on a bien $n \geqslant h_1 + h_2 + 1 \geqslant max(h_1, h_2) + 1 = h$ c.-à-d. $P(A)$ vraie

Par le principe d'induction structurelle, on a $P(A)$ vraie, $\forall A \in ArbreBin$

Démonstration

Soit $P(a) =$ "$hauteur(a) \leqslant nbssab(a)$"

1. **Base** Montrons que le propriété est vraie pour l'élément de la base : $hauteur(ABvide) = 0$ et $nbssab(ABvide) = 0$, on a bien $hauteur(ABvide) \leqslant nbssab(ABvide)$. $P(ABvide)$ est donc vraie

2. **Règle** Soient les arbres g et $d \in ArbreBin$ de hauteurs h_1, h_2 avec un nombre de sous–arbres n_1, n_2.
   Supposons que $P(g) = P(d) =$ vrai, montrons que $P(A)$ vraie quelque soit $A = (e, g, d)$:
   - pour tout arbre $A = (e, g, d)$, sa hauteur est $h = max(h_1, h_2) + 1$, et son nombre de sous-arbres non vides est $n = n_1 + n_2 + 1$
   - et comme $n_1 \geqslant h_1$ et $n_2 \geqslant h_2$
   - on a bien $n \geqslant h_1 + h_2 + 1 \geqslant max(h_1, h_2) + 1 = h$ c.-à-d. $P(A)$ vraie

Par le principe d'induction structurelle, on a $P(A)$ vraie, $\forall A \in ArbreBin$

Démonstration

Soit $P(a) =$ "$hauteur(a) \leqslant nbssab(a)$"

1. **Base** Montrons que le propriété est vraie pour l'élément de la base : $hauteur(ABvide) = 0$ et $nbssab(ABvide) = 0$, on a bien $hauteur(ABvide) \leqslant nbssab(ABvide)$. $P(ABvide)$ est donc vraie

2. **Règle** Soient les arbres g et $d \in ArbreBin$ de hauteurs h_1,h_2 avec un nombre de sous–arbres n_1, n_2.
   Supposons que $P(g) = P(d) =$ vrai, montrons que $P(A)$ vraie quelque soit $A = (e, g, d)$:
   - pour tout arbre $A = (e, g, d)$, sa hauteur est $h = max(h_1, h_2) + 1$, et son nombre de sous-arbres non vides est $n = n_1 + n_2 + 1$
   - et comme $n_1 \geqslant h_1$ et $n_2 \geqslant h_2$
   - on a bien $n \geqslant h_1 + h_2 + 1 \geqslant max(h_1, h_2) + 1 = h$ c.-à-d. $P(A)$ vraie

Par le principe d'induction structurelle, on a $P(A)$ vraie, $\forall A \in ArbreBin$

Démonstration

Soit $P(a) =$ “$hauteur(a) \leqslant nbssab(a)$”

1. **Base** Montrons que le propriété est vraie pour l'élément de la base : $hauteur(ABvide) = 0$ et $nbssab(ABvide) = 0$, on a bien $hauteur(ABvide) \leqslant nbssab(ABvide)$. $P(ABvide)$ est donc vraie

2. **Règle** Soient les arbres g et $d \in ArbreBin$ de hauteurs h_1,h_2 avec un nombre de sous–arbres n_1, n_2.
   Supposons que $P(g) = P(d) =$ vrai, montrons que $P(A)$ vraie quelque soit $A = (e, g, d)$:
   - pour tout arbre $A = (e, g, d)$, sa hauteur est $h = max(h_1, h_2) + 1$, et son nombre de sous-arbres non vides est $n = n_1 + n_2 + 1$
   - et comme $n_1 \geqslant h_1$ et $n_2 \geqslant h_2$
   - on a bien $n \geqslant h_1 + h_2 + 1 \geqslant max(h_1, h_2) + 1 = h$ c.-à-d. $P(A)$ vraie

Par le principe d'induction structurelle, on a $P(A)$ vraie, $\forall A \in ArbreBin$

Démonstration

Soit $P(a) =$ "$hauteur(a) \leqslant nbssab(a)$"

1. **Base** Montrons que le propriété est vraie pour l'élément de la base : $hauteur(ABvide) = 0$ et $nbssab(ABvide) = 0$, on a bien $hauteur(ABvide) \leqslant nbssab(ABvide)$. $P(ABvide)$ est donc vraie

2. **Règle** Soient les arbres g et $d \in ArbreBin$ de hauteurs h_1,h_2 avec un nombre de sous–arbres n_1, n_2.
   Supposons que $P(g) = P(d) =$ vrai, montrons que $P(A)$ vraie quelque soit $A = (e, g, d)$:
   - pour tout arbre $A = (e, g, d)$, sa hauteur est $h = max(h_1, h_2) + 1$, et son nombre de sous-arbres non vides est $n = n_1 + n_2 + 1$
   - et comme $n_1 \geqslant h_1$ et $n_2 \geqslant h_2$
   - on a bien $n \geqslant h_1 + h_2 + 1 \geqslant max(h_1, h_2) + 1 = h$ c.-à-d. $P(A)$ vraie

Par le principe d'induction structurelle, on a $P(A)$ vraie, $\forall A \in ArbreBin$

Démonstration

Soit $P(a) =$ “$hauteur(a) \leqslant nbssab(a)$”

1. **Base** Montrons que le propriété est vraie pour l'élément de la base : $hauteur(ABvide) = 0$ et $nbssab(ABvide) = 0$, on a bien $hauteur(ABvide) \leqslant nbssab(ABvide)$. $P(ABvide)$ est donc vraie

2. **Règle** Soient les arbres g et $d \in ArbreBin$ de hauteurs h_1,h_2 avec un nombre de sous–arbres n_1, n_2.
   Supposons que $P(g) = P(d) =$ vrai, montrons que $P(A)$ vraie quelque soit $A = (e, g, d)$:

   - pour tout arbre $A = (e, g, d)$, sa hauteur est $h = max(h_1, h_2) + 1$, et son nombre de sous-arbres non vides est $n = n_1 + n_2 + 1$
   - et comme $n_1 \geqslant h_1$ et $n_2 \geqslant h_2$
   - on a bien $n \geqslant h_1 + h_2 + 1 \geqslant max(h_1, h_2) + 1 = h$ c.-à-d. $P(A)$ vraie

Par le principe d'induction structurelle, on a $P(A)$ vraie, $\forall A \in ArbreBin$

Démonstration

Soit $P(a) =$ "$hauteur(a) \leqslant nbssab(a)$"

1. **Base** Montrons que le propriété est vraie pour l'élément de la base : $hauteur(ABvide) = 0$ et $nbssab(ABvide) = 0$, on a bien $hauteur(ABvide) \leqslant nbssab(ABvide)$. $P(ABvide)$ est donc vraie

2. **Règle** Soient les arbres g et $d \in ArbreBin$ de hauteurs h_1,h_2 avec un nombre de sous–arbres n_1, n_2.
Supposons que $P(g) = P(d) =$ vrai, montrons que $P(A)$ vraie quelque soit $A = (e, g, d)$:

    - pour tout arbre $A = (e, g, d)$, sa hauteur est $h = max(h_1, h_2) + 1$, et son nombre de sous-arbres non vides est $n = n_1 + n_2 + 1$
    - et comme $n_1 \geqslant h_1$ et $n_2 \geqslant h_2$
    - on a bien $n \geqslant h_1 + h_2 + 1 \geqslant max(h_1, h_2) + 1 = h$ c.-à-d. $P(A)$ vraie

Par le principe d'induction structurelle, on a $P(A)$ vraie, $\forall A \in ArbreBin$

Démonstration

Soit $P(a) =$ "$hauteur(a) \leqslant nbssab(a)$"

1. **Base** Montrons que le propriété est vraie pour l'élément de la base : $hauteur(ABvide) = 0$ et $nbssab(ABvide) = 0$, on a bien $hauteur(ABvide) \leqslant nbssab(ABvide)$. $P(ABvide)$ est donc vraie

2. **Règle** Soient les arbres g et $d \in ArbreBin$ de hauteurs h_1,h_2 avec un nombre de sous–arbres n_1, n_2.
Supposons que $P(g) = P(d) =$ vrai, montrons que $P(A)$ vraie quelque soit $A = (e, g, d)$:
   - pour tout arbre $A = (e, g, d)$, sa hauteur est $h = max(h_1, h_2) + 1$, et son nombre de sous-arbres non vides est $n = n_1 + n_2 + 1$
   - et comme $n_1 \geqslant h_1$ et $n_2 \geqslant h_2$
   - on a bien $n \geqslant h_1 + h_2 + 1 \geqslant max(h_1, h_2) + 1 = h$ c.-à-d. $P(A)$ vraie

Par le principe d'induction structurelle, on a $P(A)$ vraie, $\forall A \in ArbreBin$

Démonstration

Soit $P(a) =$ "$hauteur(a) \leqslant nbssab(a)$"

1. **Base** Montrons que le propriété est vraie pour l'élément de la base : $hauteur(ABvide) = 0$ et $nbssab(ABvide) = 0$, on a bien $hauteur(ABvide) \leqslant nbssab(ABvide)$. $P(ABvide)$ est donc vraie

2. **Règle** Soient les arbres g et $d \in ArbreBin$ de hauteurs h_1, h_2 avec un nombre de sous–arbres n_1, n_2.
   Supposons que $P(g) = P(d) =$ vrai, montrons que $P(A)$ vraie quelque soit $A = (e, g, d)$:
   - pour tout arbre $A = (e, g, d)$, sa hauteur est $h = max(h_1, h_2) + 1$, et son nombre de sous-arbres non vides est $n = n_1 + n_2 + 1$
   - et comme $n_1 \geqslant h_1$ et $n_2 \geqslant h_2$
   - on a bien $n \geqslant h_1 + h_2 + 1 \geqslant max(h_1, h_2) + 1 = h$ c.-à-d. $P(A)$ vraie

Par le principe d'induction structurelle, on a $P(A)$ vraie, $\forall A \in ArbreBin$

Démonstration

Soit $P(a) =$ "$hauteur(a) \leqslant nbssab(a)$"

1. **Base** Montrons que le propriété est vraie pour l'élément de la base : $hauteur(ABvide) = 0$ et $nbssab(ABvide) = 0$, on a bien $hauteur(ABvide) \leqslant nbssab(ABvide)$. $P(ABvide)$ est donc vraie

2. **Règle** Soient les arbres g et $d \in ArbreBin$ de hauteurs h_1,h_2 avec un nombre de sous–arbres n_1, n_2.
   Supposons que $P(g) = P(d) =$ vrai, montrons que $P(A)$ vraie quelque soit $A = (e, g, d)$:
   - pour tout arbre $A = (e, g, d)$, sa hauteur est $h = max(h_1, h_2) + 1$, et son nombre de sous-arbres non vides est $n = n_1 + n_2 + 1$
   - et comme $n_1 \geqslant h_1$ et $n_2 \geqslant h_2$
   - on a bien $n \geqslant h_1 + h_2 + 1 \geqslant max(h_1, h_2) + 1 = h$ c.-à-d. $P(A)$ vraie

Par le principe d'induction structurelle, on a $P(A)$ vraie, $\forall A \in ArbreBin$

Démonstration

Soit $P(a) =$ "$hauteur(a) \leqslant nbssab(a)$"

1. **Base** Montrons que le propriété est vraie pour l'élément de la base : $hauteur(ABvide) = 0$ et $nbssab(ABvide) = 0$, on a bien $hauteur(ABvide) \leqslant nbssab(ABvide)$. $P(ABvide)$ est donc vraie

2. **Règle** Soient les arbres g et $d \in ArbreBin$ de hauteurs h_1,h_2 avec un nombre de sous–arbres n_1, n_2.
   Supposons que $P(g) = P(d) =$ vrai, montrons que $P(A)$ vraie quelque soit $A = (e, g, d)$:
   - pour tout arbre $A = (e, g, d)$, sa hauteur est $h = max(h_1, h_2) + 1$, et son nombre de sous-arbres non vides est $n = n_1 + n_2 + 1$
   - et comme $n_1 \geqslant h_1$ et $n_2 \geqslant h_2$
   - on a bien $n \geqslant h_1 + h_2 + 1 \geqslant max(h_1, h_2) + 1 = h$ c.-à-d. $P(A)$ vraie

Par le principe d'induction structurelle, on a $P(A)$ vraie, $\forall A \in ArbreBin$

Démonstration

Soit $P(a) =$ "$hauteur(a) \leqslant nbssab(a)$"

1. **Base** Montrons que le propriété est vraie pour l'élément de la base : $hauteur(ABvide) = 0$ et $nbssab(ABvide) = 0$, on a bien $hauteur(ABvide) \leqslant nbssab(ABvide)$. $P(ABvide)$ est donc vraie

2. **Règle** Soient les arbres g et $d \in ArbreBin$ de hauteurs h_1,h_2 avec un nombre de sous–arbres n_1, n_2.
   Supposons que $P(g) = P(d) =$ vrai, montrons que $P(A)$ vraie quelque soit $A = (e, g, d)$:
   - pour tout arbre $A = (e, g, d)$, sa hauteur est $h = max(h_1, h_2) + 1$, et son nombre de sous-arbres non vides est $n = n_1 + n_2 + 1$
   - et comme $n_1 \geqslant h_1$ et $n_2 \geqslant h_2$
   - on a bien $n \geqslant h_1 + h_2 + 1 \geqslant max(h_1, h_2) + 1 = h$ c.-à-d. $P(A)$ vraie

Par le principe d'induction structurelle, on a $P(A)$ vraie, $\forall A \in ArbreBin$

Démonstration

Soit $P(a) =$ “$hauteur(a) \leqslant nbssab(a)$”

1. **Base** Montrons que le propriété est vraie pour l'élément de la base : $hauteur(ABvide) = 0$ et $nbssab(ABvide) = 0$, on a bien $hauteur(ABvide) \leqslant nbssab(ABvide)$. $P(ABvide)$ est donc vraie

2. **Règle** Soient les arbres g et $d \in ArbreBin$ de hauteurs h_1, h_2 avec un nombre de sous–arbres n_1, n_2.
   Supposons que $P(g) = P(d) =$ vrai, montrons que $P(A)$ vraie quelque soit $A = (e, g, d)$:
   - pour tout arbre $A = (e, g, d)$, sa hauteur est $h = max(h_1, h_2) + 1$, et son nombre de sous-arbres non vides est $n = n_1 + n_2 + 1$
   - et comme $n_1 \geqslant h_1$ et $n_2 \geqslant h_2$
   - on a bien $n \geqslant h_1 + h_2 + 1 \geqslant max(h_1, h_2) + 1 = h$ c.-à-d. $P(A)$ vraie

Par le principe d'induction structurelle, on a $P(A)$ vraie, $\forall A \in ArbreBin$

Démonstration

Soit $P(a) =$ "$hauteur(a) \leqslant nbssab(a)$"

1. **Base** Montrons que le propriété est vraie pour l'élément de la base : $hauteur(ABvide) = 0$ et $nbssab(ABvide) = 0$, on a bien $hauteur(ABvide) \leqslant nbssab(ABvide)$. $P(ABvide)$ est donc vraie

2. **Règle** Soient les arbres g et $d \in ArbreBin$ de hauteurs h_1,h_2 avec un nombre de sous–arbres n_1, n_2.
   Supposons que $P(g) = P(d) =$ vrai, montrons que $P(A)$ vraie quelque soit $A = (e, g, d)$:
   - pour tout arbre $A = (e, g, d)$, sa hauteur est $h = max(h_1, h_2) + 1$, et son nombre de sous-arbres non vides est $n = n_1 + n_2 + 1$
   - et comme $n_1 \geqslant h_1$ et $n_2 \geqslant h_2$
   - on a bien $n \geqslant h_1 + h_2 + 1 \geqslant max(h_1, h_2) + 1 = h$ c.-à-d. $P(A)$ vraie

Par le principe d'induction structurelle, on a $P(A)$ vraie, $\forall A \in ArbreBin$