Cours 1
Langages logiques
Aspects syntaxiques

Logique – Licence Informatique

SCIENCES SORBONNE UNIVERSITÉ

Langage logique

- permet d'exprimer des énoncés – des assertions – qui peuvent être « vrais » ou « faux »

Langage logique

- permet d'exprimer des énoncés – des assertions – qui peuvent être « vrais » ou « faux »
  - énoncé : exprime des propriétés sur des objets

Langage logique

- permet d'exprimer des énoncés – des assertions – qui peuvent être « vrais » ou « faux »
  - énoncé : exprime des propriétés sur des objets

    exemple :
    « $2 \leq 3$ *et* $3 = 7 + 2$ »

Langage logique

- permet d'exprimer des énoncés – des assertions – qui peuvent être « vrais » ou « faux »
  - énoncé : exprime des propriétés sur des objets

    exemple :
    « $2 \leq 3$ *et* $3 = 7 + 2$ »

    objets : 2, 3, 7, 7+2

Langage logique

- permet d'exprimer des énoncés – des assertions – qui peuvent être « vrais » ou « faux »
  - énoncé : exprime des propriétés sur des objets

    exemple :
    « $2 \leq 3$ *et* $3 = 7 + 2$ »

    objets : 2, 3, 7, 7+2
    propriétés : $\leq$, =

Langage logique

- permet d'exprimer des énoncés – des assertions – qui peuvent être « vrais » ou « faux »
  - énoncé : exprime des propriétés sur des objets

    exemple :
    « *le tableau d'entiers tab est trié dans l'ordre croissant* »

Langage logique

- permet d'exprimer des énoncés – des assertions – qui peuvent être « vrais » ou « faux »
  - énoncé : exprime des propriétés sur des objets

    exemple :
    « *le tableau d'entiers tab est trié dans l'ordre croissant* »

    objet : *tab*

Langage logique

- permet d'exprimer des énoncés – des assertions – qui peuvent être « vrais » ou « faux »
  - énoncé : exprime des propriétés sur des objets

    exemple :
    « *le tableau d'entiers tab est trié dans l'ordre croissant* »

    objet : *tab*
    propriété : *être trié dans l'ordre croissant*

Langage logique

- permet d'exprimer des énoncés – des assertions – qui peuvent être « vrais » ou « faux »
  - énoncé : exprime des propriétés sur des objets

    exemple :
    « *Si le soleil brille, alors tous les étudiants sont heureux.* »

Langage logique

- permet d'exprimer des énoncés – des assertions – qui peuvent être « vrais » ou « faux »
  - énoncé : exprime des propriétés sur des objets

    exemple :
    « *Si le soleil brille, alors tous les étudiants sont heureux.* »

    objets : *le soleil*, *les étudiants*

Langage logique

- permet d'exprimer des énoncés – des assertions – qui peuvent être « vrais » ou « faux »
  - énoncé : exprime des propriétés sur des objets

    exemple :
    « *Si le soleil brille, alors tous les étudiants sont heureux.* »

    objets : *le soleil*, *les étudiants*
    propriétés : *briller*, *être heureux*

Langage logique

- permet d'exprimer des énoncés – des assertions – qui peuvent être « vrais » ou « faux »
- les « phrases » d'un langage logique sont les **formules logiques**

Langage logique

- permet d'exprimer des énoncés – des assertions – qui peuvent être « vrais » ou « faux »
- les « phrases » d'un langage logique sont les **formules logiques** qui peuvent être obtenues à partir :

Langage logique

- permet d'exprimer des énoncés – des assertions – qui peuvent être « vrais » ou « faux »
- les « phrases » d'un langage logique sont les **formules logiques** qui peuvent être obtenues à partir :
  - d'énoncés atomiques : les **formules atomiques**

Langage logique

- permet d'exprimer des énoncés – des assertions – qui peuvent être « vrais » ou « faux »
- les « phrases » d'un langage logique sont les **formules logiques** qui peuvent être obtenues à partir :
  - d'énoncés atomiques : les **formules atomiques** (*a priori* indépendants les uns des autres)

Langage logique

- permet d'exprimer des énoncés – des assertions – qui peuvent être « vrais » ou « faux »
- les « phrases » d'un langage logique sont les **formules logiques** qui peuvent être obtenues à partir :
  - d'énoncés atomiques : les **formules atomiques** (*a priori* indépendants les uns des autres)

    exemples : $2 \leq 3$, *le soleil brille*

Langage logique

- permet d'exprimer des énoncés – des assertions – qui peuvent être « vrais » ou « faux »
- les « phrases » d'un langage logique sont les **formules logiques** qui peuvent être obtenues à partir :
  - d'énoncés atomiques : les **formules atomiques**
  - de **connecteurs** logiques (et, ou, etc.)

Langage logique

- permet d'exprimer des énoncés – des assertions – qui peuvent être « vrais » ou « faux »
- les « phrases » d'un langage logique sont les **formules logiques** qui peuvent être obtenues à partir :
  - d'énoncés atomiques : les **formules atomiques**
  - de **connecteurs** logiques (et, ou, etc.)

exemple : $2 \leq 3$ et $3 = 7 + 2$

Langage logique

- permet d'exprimer des énoncés – des assertions – qui peuvent être « vrais » ou « faux »
- les « phrases » d'un langage logique sont les **formules logiques** qui peuvent être obtenues à partir :
  - d'énoncés atomiques : les **formules atomiques**
  - de **connecteurs** logiques (et, ou, etc.)
  - de **quantificateurs** (pour-tout, il-existe)

Langage logique

- permet d'exprimer des énoncés – des assertions – qui peuvent être « vrais » ou « faux »
- les « phrases » d'un langage logique sont les **formules logiques** qui peuvent être obtenues à partir :
  - d'énoncés atomiques : les **formules atomiques**
  - de **connecteurs** logiques (et, ou, etc.)
  - de **quantificateurs** (pour-tout, il-existe)

  exemple : « *tous les étudiants sont heureux* »

Langage logique

- permet d'exprimer des énoncés – des assertions – qui peuvent être « vrais » ou « faux »
- les « phrases » d'un langage logique sont les **formules logiques**
- il existe plusieurs langages logiques offrant un pouvoir d'expression plus ou moins grand selon les constructions qu'ils permettent

Langage logique

- permet d'exprimer des énoncés – des assertions – qui peuvent être « vrais » ou « faux »
- les « phrases » d'un langage logique sont les **formules logiques**
- il existe plusieurs langages logiques offrant un pouvoir d'expression plus ou moins grand selon les constructions qu'ils permettent

  dans ce cours on distingue :

  - les langages logiques sans variable
  - les langages logiques avec variable

Logique des propositions

- les énoncés atomiques sont désignés par des **symboles de proposition**

Logique des propositions

- les énoncés atomiques sont désignés par des **symboles de proposition**
  - $\mathcal{P}_0$: ensemble des symboles de proposition

Logique des propositions

- les énoncés atomiques sont désignés par des **symboles de proposition**
  - $\mathcal{P}_0$: ensemble des symboles de proposition

exemples :

symbole	énoncé
p_0	*Il pleut*
p_1	*Socrate suit un cours de logique*
p_2	*Socrate lit ses messages sur son smartphone*
p_3	*Socrate comprend la logique*
p_4	$2 \leq 5$
p_5	$3 = 2 + 5$

Logique des propositions

- les énoncés atomiques sont désignés par des **symboles de proposition**
  - $\mathcal{P}_0$: ensemble des symboles de proposition

exemples :

symbole	énoncé
p_0	*Il pleut*
p_1	*Socrate suit un cours de logique*
p_2	*Socrate lit ses messages sur son smartphone*
p_3	*Socrate comprend la logique*
p_4	$2 \leq 5$
p_5	$3 = 2 + 5$

$\mathcal{P}_0 = \{p_0, p_1, p_2, p_3, p_4, p_5\}$

Logique des propositions

- les énoncés atomiques sont désignés par des **symboles de proposition**
  - $\mathcal{P}_0$: ensemble des symboles de proposition

exemples :

symbole	énoncé
p_0	*Il pleut*
p_1	*Socrate suit un cours de logique*
p_2	*Socrate lit ses messages sur son smartphone*
p_3	*Socrate comprend la logique*
p_4	$2 \leq 5$
p_5	$3 = 2 + 5$

  - énoncés *a priori* indépendants : Socrate peut très bien suivre un cours de logique sans lire ses messages sur son smartphone, comme il n'est pas impossible qu'il comprenne la logique en lisant ses messages sur son smartphone

Logique des propositions

- les énoncés atomiques sont désignés par des **symboles de proposition**
  - $\mathcal{P}_0$: ensemble des symboles de proposition
- *exemple :* $my\_tab =$ $\boxed{2 \mid 5 \mid 6 \mid 9}$

Logique des propositions

- les énoncés atomiques sont désignés par des **symboles de proposition**
  - $\mathcal{P}_0$: ensemble des symboles de proposition
- *exemple :* $my\_tab = \boxed{2 \mid 5 \mid 6 \mid 9}$

« **le** *tableau d'entiers my_tab est trié par ordre croissant* »

Logique des propositions

- les énoncés atomiques sont désignés par des **symboles de proposition**
  - $\mathcal{P}_0$: ensemble des symboles de proposition
- *exemple :* $my\_tab =$

2	5	6	9

« **le** *tableau d'entiers my_tab est trié par ordre croissant* »

  - énoncé atomique représenté par un symbole p

Logique des propositions

- les énoncés atomiques sont désignés par des **symboles de proposition**
  - $\mathcal{P}_0$: ensemble des symboles de proposition
- *exemple :* $my_tab = $ | 2 | 5 | 6 | 9 |

« **le** *tableau d'entiers my_tab est trié par ordre croissant* »

  - énoncé atomique représenté par un symbole p

    p désigne l'énoncé « **le** *tableau d'entiers my_tab est trié par ordre croissant* »

Logique des propositions

- les énoncés atomiques sont désignés par des **symboles de proposition**
  - $\mathcal{P}_0$: ensemble des symboles de proposition
- *exemple :* $my_tab = $ $\boxed{2}\boxed{5}\boxed{6}\boxed{9}$

« **le** *tableau d'entiers my_tab est trié par ordre croissant* »

- énoncé atomique représenté par un symbole p

  p désigne l'énoncé « **le** *tableau d'entiers my_tab est trié par ordre croissant* »

  $\mathcal{P}_0 = \{p\}$

Logique des propositions

- les énoncés atomiques sont désignés par des **symboles de proposition**
  - $\mathcal{P}_0$: ensemble des symboles de proposition
- *exemple :* $my_tab = $ $\boxed{2 \mid 5 \mid 6 \mid 9}$

« **le** *tableau d'entiers my_tab est trié par ordre croissant* »

  - énoncé représenté par une conjonction d'énoncés atomiques

Logique des propositions

- les énoncés atomiques sont désignés par des **symboles de proposition**
  - $\mathcal{P}_0$: ensemble des symboles de proposition
- *exemple :* $my_tab = $ $\boxed{2 \mid 5 \mid 6 \mid 9}$

« **le** *tableau d'entiers my_tab est trié par ordre croissant* »

- énoncé représenté par une conjonction d'énoncés atomiques

p_1	*l'élément* 2 *est inférieur ou égal à l'élément* 5
p_2	*l'élément* 5 *est inférieur ou égal à l'élément* 6
p_3	*l'élément* 6 *est inférieur ou égal à l'élément* 9

Logique des propositions

- les énoncés atomiques sont désignés par des **symboles de proposition**
  - $\mathcal{P}_0$: ensemble des symboles de proposition
- *exemple :* $my\_tab = \boxed{2 \mid 5 \mid 6 \mid 9}$

« **le** *tableau d'entiers my_tab est trié par ordre croissant* »

- énoncé représenté par une conjonction d'énoncés atomiques

p_1	*l'élément* 2 *est inférieur ou égal à l'élément* 5
p_2	*l'élément* 5 *est inférieur ou égal à l'élément* 6
p_3	*l'élément* 6 *est inférieur ou égal à l'élément* 9

$\mathcal{P}_0 = \{p_1, p_2, p_3\}$

Logique des propositions

- les énoncés atomiques sont désignés par des **symboles de proposition**
  - $\mathcal{P}_0$: ensemble des symboles de proposition
- *exemple :* $my\_tab = \boxed{2 \mid 5 \mid 6 \mid 9}$

« **le** *tableau d'entiers my_tab est trié par ordre croissant* »

- énoncé représenté par une conjonction d'énoncés atomiques

p_1	*l'élément* 2 *est inférieur ou égal à l'élément* 5
p_2	*l'élément* 5 *est inférieur ou égal à l'élément* 6
p_3	*l'élément* 6 *est inférieur ou égal à l'élément* 9

énoncés atomiques

$$\begin{array}{ccccc} \uparrow & & \uparrow & & \uparrow \\ p_1 & \text{et} & p_2 & \text{et} & p_3 \\ & \downarrow & & \downarrow & \end{array}$$

connecteur logique « et »

Logique des propositions

- les énoncés atomiques sont désignés par des **symboles de proposition**
  - $\mathcal{P}_0$: ensemble des symboles de proposition
- $\mathcal{P}_0$: langage des énoncés atomiques

Logique des propositions

- les énoncés atomiques sont désignés par des **symboles de proposition**
  - $\mathcal{P}_0$: ensemble des symboles de proposition
- $\mathcal{P}_0$: langage des énoncés atomiques

symbole	énoncé
p_1	*Socrate suit un cours de logique*
p_4	$2 \leq 5$
p_5	$3 = 2 + 5$

Logique des propositions

- les énoncés atomiques sont désignés par des **symboles de proposition**
  - $\mathcal{P}_0$: ensemble des symboles de proposition
- $\mathcal{P}_0$: langage des énoncés atomiques

symbole	énoncé
p_1	*Socrate suit un cours de logique*
p_4	$2 \leq 5$
p_5	$3 = 2 + 5$

  - on ne prend pas en compte que l'énoncé *Socrate suit un cours de logique* peut se décomposer en un sujet (*Socrate*) et un prédicat (*suit un cours de logique*)

Logique des propositions

- les énoncés atomiques sont désignés par des **symboles de proposition**
  - $\mathcal{P}_0$: ensemble des symboles de proposition
- $\mathcal{P}_0$: langage des énoncés atomiques

symbole	énoncé
p_1	*Socrate suit un cours de logique*
p_4	$2 \leq 5$
p_5	$3 = 2 + 5$

  - on ne prend pas en compte que l'énoncé *Socrate suit un cours de logique* peut se décomposer en un sujet (*Socrate*) et un prédicat (*suit un cours de logique*)
  - 5 ne désigne pas nécessairement le même objet dans les énoncés $2 \leq 5$ et $3 = 2 + 5$

Logique des propositions

- les énoncés atomiques sont désignés par des **symboles de proposition**
  - $\mathcal{P}_0$: ensemble des symboles de proposition
- $\mathcal{P}_0$: langage des énoncés atomiques

symbole	énoncé
p_1	*Socrate suit un cours de logique*
p_4	$2 \leq 5$
p_5	$3 = 2 + 5$

  - on ne prend pas en compte que l'énoncé *Socrate suit un cours de logique* peut se décomposer en un sujet (*Socrate*) et un prédicat (*suit un cours de logique*)
  - 5 ne désigne pas nécessairement le même objet dans les énoncés $2 \leq 5$ et $3 = 2 + 5$
  - la « valeur de vérité » de $2 \leq 5$ et $3 = 2 + 5$ ne dépend pas de l'interprétation de 2 , 3 , 5 , $\leq$ et $+$ ... on peut seulement décider que ces énoncés sont « vrais » ou « faux »

Logique des propositions

- les énoncés atomiques sont désignés par des **symboles de proposition**
  - $\mathcal{P}_0$: ensemble des symboles de proposition
- $\mathcal{P}_0$: langage des énoncés atomiques
  - enrichir le langage des énoncés atomiques :
    - définition d'un langage pour désigner les objets de l'univers du discours : langage de termes
    - introduction d'un ensemble de symboles de prédicat pour désigner les propriétés

Logique des prédicats

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **objets** sur lesquels il s'applique

Logique des prédicats

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **objets** sur lesquels il s'applique
  - $\mathcal{P}$: ensemble des symboles de prédicat

Logique des prédicats

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **objets** sur lesquels il s'applique
  - $\mathcal{P}$: ensemble des symboles de prédicat
- *exemple :* « *Socrate suit un cours de logique* »

Logique des prédicats

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **objets** sur lesquels il s'applique
  - $\mathcal{P}$: ensemble des symboles de prédicat
- *exemple :* « *Socrate suit un cours de logique* »
  - peut se décomposer en :

    un objet : *Socrate*

Logique des prédicats

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **objets** sur lesquels il s'applique
  - $\mathcal{P}$: ensemble des symboles de prédicat
- *exemple :* « *Socrate suit un cours de logique* »
  - peut se décomposer en :

    un objet : *Socrate*
    un prédicat unaire : *suit_un_cours_de_logique*

Logique des prédicats

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **objets** sur lesquels il s'applique
  - $\mathcal{P}$: ensemble des symboles de prédicat
- *exemple :* « *Socrate suit un cours de logique* »
  - peut se décomposer en :

    un objet : *Socrate*
    un prédicat unaire : *suit_un_cours_de_logique*
    formule atomique : *suit_un_cours_de_logique(Socrate)*

Logique des prédicats

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **objets** sur lesquels il s'applique
  - $\mathcal{P}$: ensemble des symboles de prédicat
- *exemple :* « *Socrate suit un cours de logique* »
  - peut se décomposer en :

    un objet : *Socrate*
    un prédicat unaire : *suit_un_cours_de_logique*
    formule atomique : *suit_un_cours_de_logique(Socrate)*

    $\mathcal{P}_1$: sous-ensemble de $\mathcal{P}$ contenant les symboles de prédicat unaire

Logique des prédicats

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **objets** sur lesquels il s'applique
  - $\mathcal{P}$: ensemble des symboles de prédicat
- *exemple :* « $2 \leq 5$ »

Logique des prédicats

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **objets** sur lesquels il s'applique
  - $\mathcal{P}$: ensemble des symboles de prédicat
- *exemple :* « $2 \leq 5$ »
  - peut se décomposer en :

    un prédicat binaire : $\leq$

Logique des prédicats

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **objets** sur lesquels il s'applique
  - $\mathcal{P}$: ensemble des symboles de prédicat
- *exemple :* « $2 \leq 5$ »
  - peut se décomposer en :

    un prédicat binaire : $\leq$
    appliqué sur les objets 2 et 5

Logique des prédicats

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **objets** sur lesquels il s'applique
  - $\mathcal{P}$: ensemble des symboles de prédicat
- *exemple :* « $2 \leq 5$ »
  - peut se décomposer en :

    un prédicat binaire : $\leq$
    appliqué sur les objets 2 et 5
    formule atomique : $\leq(2, 5)$

Logique des prédicats

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **objets** sur lesquels il s'applique
  - $\mathcal{P}$: ensemble des symboles de prédicat
- *exemple :* « $2 \leq 5$ »
  - peut se décomposer en :

    un prédicat binaire : $\leq$
    appliqué sur les objets 2 et 5
    formule atomique : $\leq (2, 5)$

    $\mathcal{P}_2$: sous-ensemble de $\mathcal{P}$ contenant les symboles de prédicat binaire

Logique des prédicats

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **objets** sur lesquels il s'applique
  - $\mathcal{P}$: ensemble des symboles de prédicat
- *exemple :* $my\_tab =$ | 2 | 5 | 6 | 9 |

  « **le** *tableau d'entiers my_tab est trié par ordre croissant* »

Logique des prédicats

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **objets** sur lesquels il s'applique
  - $\mathcal{P}$: ensemble des symboles de prédicat
- *exemple :* $my_tab = $ | 2 | 5 | 6 | 9 |

  « **le** *tableau d'entiers my_tab est trié par ordre croissant* »
  - peut se décomposer en :

    une conjonction de formules atomiques

Logique des prédicats

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **objets** sur lesquels il s'applique
  - $\mathcal{P}$: ensemble des symboles de prédicat
- *exemple :* $my\_tab = \boxed{2} \boxed{5} \boxed{6} \boxed{9}$

  « **le** *tableau d'entiers my_tab est trié par ordre croissant* »
  - peut se décomposer en :

    une conjonction de formules atomiques
    composées de l'application du prédicat binaire $\leq$

Logique des prédicats

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **objets** sur lesquels il s'applique
  - $\mathcal{P}$: ensemble des symboles de prédicat
- *exemple :* $my\_tab = \boxed{2}\boxed{5}\boxed{6}\boxed{9}$

  « **le** *tableau d'entiers my_tab est trié par ordre croissant* »
  - peut se décomposer en :

    une conjonction de formules atomiques
    composées de l'application du prédicat binaire $\leq$
    sur les objets 2 et 5, 5 et 6, et 6 et 9

Logique des prédicats

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **objets** sur lesquels il s'applique
  - $\mathcal{P}$: ensemble des symboles de prédicat
- *exemple :* $my\_tab =$ $\boxed{2}\boxed{5}\boxed{6}\boxed{9}$

  « **le** *tableau d'entiers my_tab est trié par ordre croissant* »
  - peut se décomposer en :

    une conjonction de formules atomiques
    composées de l'application du prédicat binaire $\leq$
    sur les objets 2 et 5, 5 et 6, et 6 et 9
    formule logique :

$$\underbrace{\leq(2,5)}_{\text{formules atomiques}} \ \underbrace{\text{et}}_{\text{connecteur logique « et »}} \ \leq(5,6) \ \text{et} \ \leq(6,9)$$

Logique des prédicats

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **objets** sur lesquels il s'applique
  - $\mathcal{P}$: ensemble des symboles de prédicat
- *exemple :* $my\_tab =$ $\boxed{2 \mid 5 \mid 6 \mid 9}$

  « **le** *tableau d'entiers my_tab est trié par ordre croissant* »
  - **formule logique** :$\leq(2,5)$ et $\leq(5,6)$ et $\leq(6,9)$

Logique des prédicats

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **objets** sur lesquels il s'applique
  - $\mathcal{P}$: ensemble des symboles de prédicat
- *exemple :* $my\_tab =$

2	5	6	9

  « **le** *tableau d'entiers my_tab est trié par ordre croissant* »
  - **formule logique** :$\leq (2,5)$ et $\leq (5,6)$ et $\leq (6,9)$
  - ne pas dépendre des valeurs particulières 2, 5, 6 et 9 ?

Logique des prédicats

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **objets** sur lesquels il s'applique
  - $\mathcal{P}$: ensemble des symboles de prédicat
- *exemple :* $my\_tab =$ | 2 | 5 | 6 | 9 |

  « **le** *tableau d'entiers my_tab est trié par ordre croissant* »
  - **formule logique** : $\leq (2,5)$ et $\leq (5,6)$ et $\leq (6,9)$
  - ne pas dépendre des valeurs particulières 2, 5, 6 et 9 ?
  - « **un** *tableau t_4 contenant 4 entiers est trié par ordre croissant* »

Logique des prédicats

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **objets** sur lesquels il s'applique
  - $\mathcal{P}$: ensemble des symboles de prédicat
- *exemple :* $my\_tab = \boxed{2 \mid 5 \mid 6 \mid 9}$

  « **le** *tableau d'entiers my_tab est trié par ordre croissant* »
  - **formule logique** : $\leq(2,5)$ et $\leq(5,6)$ et $\leq(6,9)$
  - ne pas dépendre des valeurs particulières 2, 5, 6 et 9 ?
  - « **un** *tableau t_4 contenant 4 entiers est trié par ordre croissant* »
  - expliciter le lien entre le tableau *my_tab* et les éléments 2, 5, 6 et 9

Logique des prédicats

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **objets** sur lesquels il s'applique
  - $\mathcal{P}$: ensemble des symboles de prédicat
- *exemple :*

$$t_4 = \boxed{?\,|\,?\,|\,?\,|\,?}$$

« **un** *tableau t_4 contenant 4 entiers est trié par ordre croissant* »

Logique des prédicats

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **objets** sur lesquels il s'applique
  - $\mathcal{P}$: ensemble des symboles de prédicat
- *exemple :*

$$t_4 = \begin{array}{cccc} 0 & 1 & 2 & 3 \\ \hline \multicolumn{1}{|c|}{pos(t_4, 0)} & \multicolumn{1}{c|}{pos(t_4, 1)} & \multicolumn{1}{c|}{pos(t_4, 2)} & \multicolumn{1}{c|}{pos(t_4, 3)} \\ \hline \end{array}$$

« **un** *tableau t_4 contenant 4 entiers est trié par ordre croissant* »

Logique des prédicats

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **objets** sur lesquels il s'applique
  - $\mathcal{P}$: ensemble des symboles de prédicat
- *exemple :*

	0	1	2	3
$t_4 =$	$pos(t_4, 0)$	$pos(t_4, 1)$	$pos(t_4, 2)$	$pos(t_4, 3)$

« **un** *tableau* t_4 *contenant 4 entiers est trié par ordre croissant* »

  - pos est un symbole de fonction binaire

Logique des prédicats

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **objets** sur lesquels il s'applique
  - $\mathcal{P}$: ensemble des symboles de prédicat
- *exemple :*

	0	1	2	3
$t_4 =$	$pos(t_4, 0)$	$pos(t_4, 1)$	$pos(t_4, 2)$	$pos(t_4, 3)$

« **un** *tableau t_4 contenant 4 entiers est trié par ordre croissant* »

  - *pos* est un symbole de fonction binaire
    - qui s'applique sur deux objets : un tableau (ici t_4) et un entier (ici 0, 1, 2 et 3)

Logique des prédicats

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **objets** sur lesquels il s'applique
  - $\mathcal{P}$: ensemble des symboles de prédicat
- *exemple :*

$$t_4 = \begin{array}{c|c|c|c} 0 & 1 & 2 & 3 \\ \hline \multicolumn{1}{|c|}{pos(t_4, 0)} & pos(t_4, 1) & pos(t_4, 2) & \multicolumn{1}{c|}{pos(t_4, 3)} \\ \hline \end{array}$$

« <u>**un**</u> *tableau t_4 contenant 4 entiers est trié par ordre croissant* »

  - *pos* est un symbole de fonction binaire
    - qui s'applique sur deux objets : un tableau (ici t_4) et un entier (ici 0, 1, 2 et 3)
    - permet de désigner un objet : $pos(t, i)$ est le i-ème élément du tableau t

Logique des prédicats

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **objets** sur lesquels il s'applique
  - $\mathcal{P}$: ensemble des symboles de prédicat
- *exemple :*

$$t_4 = \begin{array}{c|c|c|c} 0 & 1 & 2 & 3 \\ \hline pos(t_4, 0) & pos(t_4, 1) & pos(t_4, 2) & pos(t_4, 3) \end{array}$$

« **un** *tableau t_4 contenant 4 entiers est trié par ordre croissant* »

  - *pos* est un symbole de fonction binaire
    - qui s'applique sur deux objets : un tableau (ici t_4) et un entier (ici 0, 1, 2 et 3)
    - permet de désigner un objet : $pos(t, i)$ est le i-ième élément du tableau t

$$\leq (pos(t_4, 0), pos(t_4, 1)) \text{ et } \leq (pos(t_4, 1), pos(t_4, 2)) \text{ et } \leq (pos(t_4, 2), pos(t_4, 3))$$

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : les **termes**

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : les **termes**
- les termes peuvent être :
  - des noms d'objets/individus Socrate, t_4, 2, 3, 5 : **les constantes**

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : les **termes**
- les termes peuvent être :
  - des noms d'objets/individus Socrate, t_4, 2, 3, 5 : **les constantes**
  - des **fonctions** appliquées
    - à des constantes $+(2, 5)$, $pos(t_4, 0)$

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : les **termes**
- les termes peuvent être :
  - des noms d'objets/individus Socrate, t_4, 2, 3, 5 : **les constantes**
  - des **fonctions** appliquées
    - ⋆ à des constantes $+(2,5)$, $pos(t_4,0)$
    - ⋆ ou plus généralement des fonctions appliquées à d'autres termes $+(+(2,5),3)$, $pos(t_4,+(0,1))$

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : les **termes**
- les termes peuvent être :
  - des noms d'objets/individus Socrate, t_4, 2, 3, 5 : **les constantes**
  - des **fonctions** appliquées
    - à des constantes $+(2,5)$, $pos(t_4, 0)$
    - ou plus généralement des fonctions appliquées à d'autres termes $+(+(2,5),3)$, $pos(t_4, +(0,1))$
- définition inductive des termes

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : les **termes**
  - $\mathcal{F}$: ensemble de

    symboles de constante
    symboles de fonction

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : les **termes**
  - $\mathcal{F}$: ensemble de
    symboles de constante
    symboles de fonction
- *exemple :* entiers de Peano

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : les **termes**
  - $\mathcal{F}$: ensemble de
    symboles de constante
    symboles de fonction
- *exemple :* entiers de Peano
  - Z (pour zéro) est un entier de Peano

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : les **termes**
  - $\mathcal{F} = \mathcal{F}_0$: ensemble de symboles de constantes et de fonction
    $\mathcal{F}_0 = \{Z\}$ contient les symboles de constante
    symboles de fonction
- *exemple :* entiers de Peano
  - Z (pour zéro) est un entier de Peano
    - ⋆ Z est un symbole de constante (c'est un terme)

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : les **termes**
  - $\mathcal{F} = \mathcal{F}_0$: ensemble de symboles de constantes et de fonction
    $\mathcal{F}_0 = \{Z\}$ contient les symboles de constante
    symboles de fonction
- *exemple :* entiers de Peano
  - Z (pour zéro) est un entier de Peano
    - ⋆ Z est un symbole de constante (c'est un terme)
  - si n est un entier de Peano, alors $S(n)$ (le successeur de n) est un entier de Peano

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : les **termes**
  - $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_1$: ensemble de symboles de constantes et de fonction
    $\mathcal{F}_0 = \{Z\}$ contient les symboles de constante
    $\mathcal{F}_1 = \{S\}$ contient les symboles de fonction d'arité 1
- *exemple :* entiers de Peano
  - Z (pour zéro) est un entier de Peano
    - ⋆ Z est un symbole de constante (c'est un terme)
  - si n est un entier de Peano, alors $S(n)$ (le successeur de n) est un entier de Peano
    - ⋆ S est un symbole de fonction d'arité 1 (il s'applique sur un terme pour construire le terme $S(n)$)

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : les **termes**
  - $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_1$: ensemble de symboles de constantes et de fonction
    $\mathcal{F}_0 = \{Z\}$ contient les symboles de constante
    $\mathcal{F}_1 = \{S\}$ contient les symboles de fonction d'arité 1
- *exemple :* entiers de Peano

0	1	2	$\cdots$	n	$\cdots$
Z	S	S		S	
	$\vert$	$\vert$		$\vert$	
	Z	S		S	
		$\vert$		$\vert$	
		Z		$\vdots$	
				S	
				$\vert$	
				Z	

– arbre dont chaque nœud a au plus un fils
– constante Z : feuille de l'arbre
– chaque nœud étiqueté par S
a exactement 1 fils

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : les **termes**
  - $\mathcal{F}$: ensemble de

    symboles de constante
    symboles de fonction
- *exemple :* expressions arithmétiques (simples)

Langage de termes

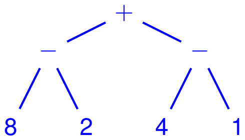
- les énoncés expriment des propriétés sur les objets de l'univers du discours : les **termes**
  - $\mathcal{F} = \mathcal{F}_0$: ensemble de symboles de constantes et de fonction
    $\mathcal{F}_0 = \mathbb{Z}$ contient les symboles de constante
    symboles de fonction
- *exemple :* expressions arithmétiques (simples)
  - les entiers relatifs sont des expressions arithmétiques
    - si $k \in \mathbb{Z}$, alors k est un terme (constante)

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : les **termes**
  - $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_2$: ensemble de symboles de constantes et de fonction
    $\mathcal{F}_0 = \mathbb{Z}$ contient les symboles de constante
    $\mathcal{F}_2 = \{+, -, \times, /\}$ contient les symboles de fonction d'arité 2
- *exemple :* expressions arithmétiques (simples)
  - les entiers relatifs sont des expressions arithmétiques
    - ★ si $k \in \mathbb{Z}$, alors k est un terme (constante)
  - si e_1 et e_2 sont des expressions arithmétiques, alors $e_1 + e_2$, $e_1 - e_2$, $e_1 \times e_2$ et e_1/e_2 sont des expressions arithmétiques

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : les **termes**
  - $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_2$: ensemble de symboles de constantes et de fonction
    $\mathcal{F}_0 = \mathbb{Z}$ contient les symboles de constante
    $\mathcal{F}_2 = \{+, -, \times, /\}$ contient les symboles de fonction d'arité 2
- *exemple :* expressions arithmétiques (simples)



$(8 - 2) + (4 - 1)$

– arbre dont chaque nœud a 0 ou 2 fils
– feuilles de l'arbre : constantes dans $\mathbb{Z}$
– chaque nœud étiqueté par $+$, $-$, $\times$ et $/$ a exactement 2 fils

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : les **termes**
  - $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_1 \cup \mathcal{F}_2 \cup \mathcal{F}_3 \cup \cdots = \bigcup_{n \geq 0} \mathcal{F}_n$ ensemble de symboles
    $\mathcal{F}_0$ contient les symboles de constante
    $\mathcal{F}_n$ contient les symboles de fonction d'arité n
- **définition inductive de l'ensemble $\mathcal{T}_0(\mathcal{F})$ des termes sans variable**

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : les **termes**
  - $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_1 \cup \mathcal{F}_2 \cup \mathcal{F}_3 \cup \cdots = \bigcup_{n \geq 0} \mathcal{F}_n$ ensemble de symboles
    $\mathcal{F}_0$ contient les symboles de constante
    $\mathcal{F}_n$ contient les symboles de fonction d'arité n
- **définition inductive de l'ensemble $\mathcal{T}_0(\mathcal{F})$ des termes sans variable**
  - toute constante $k \in \mathcal{F}_0$ est un terme dans $\mathcal{T}_0(\mathcal{F})$

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : les **termes**
  - $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_1 \cup \mathcal{F}_2 \cup \mathcal{F}_3 \cup \cdots = \bigcup_{n \geq 0} \mathcal{F}_n$ ensemble de symboles
    $\mathcal{F}_0$ contient les symboles de constante
    $\mathcal{F}_n$ contient les symboles de fonction d'arité n
- **définition inductive de l'ensemble $\mathcal{T}_0(\mathcal{F})$ des termes sans variable**
  - toute constante $k \in \mathcal{F}_0$ est un terme dans $\mathcal{T}_0(\mathcal{F})$
  - si $f \in \mathcal{F}_n$ et si $t_1, \cdots, t_n$ sont des termes dans $\mathcal{T}_0(\mathcal{F})$, alors $f(t_1, \cdots, t_n)$ est un terme dans $\mathcal{T}_0(\mathcal{F})$

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : les **termes**
  - $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_1 \cup \mathcal{F}_2 \cup \mathcal{F}_3 \cup \cdots = \bigcup_{n \geq 0} \mathcal{F}_n$ ensemble de symboles
    $\mathcal{F}_0$ contient les symboles de constante
    $\mathcal{F}_n$ contient les symboles de fonction d'arité n
- **définition inductive de l'ensemble $\mathcal{T}_0(\mathcal{F})$ des termes sans variable**
  - toute constante $k \in \mathcal{F}_0$ est un terme dans $\mathcal{T}_0(\mathcal{F})$
  - si $f \in \mathcal{F}_n$ et si $t_1, \cdots, t_n$ sont des termes dans $\mathcal{T}_0(\mathcal{F})$, alors $f(t_1, \cdots, t_n)$ est un terme dans $\mathcal{T}_0(\mathcal{F})$
- un terme $t \in \mathcal{T}_0(\mathcal{F})$ peut être vu comme un **arbre**

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : les **termes**
  - $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_1 \cup \mathcal{F}_2 \cup \mathcal{F}_3 \cup \cdots = \bigcup_{n \geq 0} \mathcal{F}_n$ ensemble de symboles
    $\mathcal{F}_0$ contient les symboles de constante
    $\mathcal{F}_n$ contient les symboles de fonction d'arité n
- **définition inductive de l'ensemble $\mathcal{T}_0(\mathcal{F})$ des termes sans variable**
  - toute constante $k \in \mathcal{F}_0$ est un terme dans $\mathcal{T}_0(\mathcal{F})$
  - si $f \in \mathcal{F}_n$ et si $t_1, \cdots, t_n$ sont des termes dans $\mathcal{T}_0(\mathcal{F})$, alors $f(t_1, \cdots, t_n)$ est un terme dans $\mathcal{T}_0(\mathcal{F})$
- un terme $t \in \mathcal{T}_0(\mathcal{F})$ peut être vu comme un **arbre**
  - les feuilles sont étiquetées par des éléments de $\mathcal{F}_0$

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : les **termes**
  - $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_1 \cup \mathcal{F}_2 \cup \mathcal{F}_3 \cup \cdots = \bigcup_{n \geq 0} \mathcal{F}_n$ ensemble de symboles
    $\mathcal{F}_0$ contient les symboles de constante
    $\mathcal{F}_n$ contient les symboles de fonction d'arité n
- **définition inductive de l'ensemble $\mathcal{T}_0(\mathcal{F})$ des termes sans variable**
  - toute constante $k \in \mathcal{F}_0$ est un terme dans $\mathcal{T}_0(\mathcal{F})$
  - si $f \in \mathcal{F}_n$ et si $t_1, \cdots, t_n$ sont des termes dans $\mathcal{T}_0(\mathcal{F})$, alors $f(t_1, \cdots, t_n)$ est un terme dans $\mathcal{T}_0(\mathcal{F})$
- un terme $t \in \mathcal{T}_0(\mathcal{F})$ peut être vu comme un **arbre**
  - les feuilles sont étiquetées par des éléments de $\mathcal{F}_0$
  - chaque nœud étiqueté par un élément de $\mathcal{F}_n$ a exactement n fils

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : les **termes**
  - $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_1 \cup \mathcal{F}_2 \cup \mathcal{F}_3 \cup \cdots = \bigcup_{n \geq 0} \mathcal{F}_n$ ensemble de symboles
    $\mathcal{F}_0$ contient les symboles de constante
    $\mathcal{F}_n$ contient les symboles de fonction d'arité n
- **définition inductive de l'ensemble $\mathcal{T}_0(\mathcal{F})$ des termes sans variable**
  - toute constante $k \in \mathcal{F}_0$ est un terme dans $\mathcal{T}_0(\mathcal{F})$
  - si $f \in \mathcal{F}_n$ et si $t_1, \cdots, t_n$ sont des termes dans $\mathcal{T}_0(\mathcal{F})$, alors $f(t_1, \cdots, t_n)$ est un terme dans $\mathcal{T}_0(\mathcal{F})$

  les définitions des entiers de Peano et des expressions arithmétiques sont des **particularisations** de la définition générale de l'ensemble $\mathcal{T}_0(\mathcal{F})$ des termes (en prenant en compte l'ensemble $\mathcal{F}$)

Langage des formules atomiques

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **termes** sur lesquels il s'applique
  - $\mathcal{P}$: ensemble des symboles de prédicat

Langage des formules atomiques

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **termes** sur lesquels il s'applique
  - $\mathcal{P}$: ensemble des symboles de prédicat

- *exemple :* « *il pleut* »

Langage des formules atomiques

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **termes** sur lesquels il s'applique
  - $\mathcal{P}$: ensemble des symboles de prédicat

- *exemple :* « *il pleut* »
  - prédicat *il_pleut* sans argument

Langage des formules atomiques

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **termes** sur lesquels il s'applique
  - $\mathcal{P} = \mathcal{P}_0$: ensemble des symboles de prédicat
    $\mathcal{P}_0$ contient les symboles de proposition

- *exemple :* « *il pleut* »
  - prédicat *il_pleut* sans argument
  - symbole de proposition (prédicat d'arité 0)

Langage des formules atomiques

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **termes** sur lesquels il s'applique
  - $\mathcal{P} = \mathcal{P}_0$: ensemble des symboles de prédicat
    $\mathcal{P}_0$ contient les symboles de proposition

- *exemple :* « *Socrate suit un cours de logique* »

Langage des formules atomiques

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **termes** sur lesquels il s'applique
  - $\mathcal{P} = \mathcal{P}_0 \cup \mathcal{P}_1$: ensemble des symboles de prédicat
    $\mathcal{P}_0$ contient les symboles de proposition
    $\mathcal{P}_1$ contient les symboles de prédicat d'arité 1

- *exemple :* « *Socrate suit un cours de logique* »
  - prédicat d'arité 1 : *suit_un_cours_de_logique*

Langage des formules atomiques

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **termes** sur lesquels il s'applique
  - $\mathcal{P} = \mathcal{P}_0 \cup \mathcal{P}_1$: ensemble des symboles de prédicat
    $\mathcal{P}_0$ contient les symboles de proposition
    $\mathcal{P}_1$ contient les symboles de prédicat d'arité 1
- *exemple :* « $2 \leq 3 + 9$ »

Langage des formules atomiques

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **termes** sur lesquels il s'applique
  - $\mathcal{P} = \mathcal{P}_0 \cup \mathcal{P}_1 \cup \mathcal{P}_2$: ensemble des symboles de prédicat

    $\mathcal{P}_0$ contient les symboles de proposition
    $\mathcal{P}_1$ contient les symboles de prédicat d'arité 1
    $\mathcal{P}_2$ contient les symboles de prédicat d'arité 2
- *exemple :* « $2 \leq 3 + 9$ »
  - prédicat d'arité 2 : $\leq$

Langage des formules atomiques

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **termes** sur lesquels il s'applique
  - $\mathcal{P} = \mathcal{P}_0 \cup \mathcal{P}_1 \cup \mathcal{P}_2 \cup \cdots$: ensemble des symboles de prédicat

    $\mathcal{P}_0$ contient les symboles de proposition

    $\mathcal{P}_n$ contient les symboles de prédicat d'arité n

Langage des formules atomiques

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **termes** sur lesquels il s'applique
  - $\mathcal{P} = \mathcal{P}_0 \cup \mathcal{P}_1 \cup \mathcal{P}_2 \cup \cdots$: ensemble des symboles de prédicat
    $\mathcal{P}_0$ contient les symboles de proposition
    $\mathcal{P}_n$ contient les symboles de prédicat d'arité n
- ensemble $\mathcal{L}_0(\mathcal{F}, \mathcal{P})$ des **formules atomiques sans variable**

Langage des formules atomiques

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **termes** sur lesquels il s'applique
  - $\mathcal{P} = \mathcal{P}_0 \cup \mathcal{P}_1 \cup \mathcal{P}_2 \cup \cdots$: ensemble des symboles de prédicat
    $\mathcal{P}_0$ contient les symboles de proposition
    $\mathcal{P}_n$ contient les symboles de prédicat d'arité n
- ensemble $\mathcal{L}_0(\mathcal{F}, \mathcal{P})$ des **formules atomiques sans variable**
  - les symboles de $\mathcal{P}_0$ sont des formules atomiques de $\mathcal{L}_0(\mathcal{F}, \mathcal{P})$

Langage des formules atomiques

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **termes** sur lesquels il s'applique
  - $\mathcal{P} = \mathcal{P}_0 \cup \mathcal{P}_1 \cup \mathcal{P}_2 \cup \cdots$: ensemble des symboles de prédicat
    $\mathcal{P}_0$ contient les symboles de proposition
    $\mathcal{P}_n$ contient les symboles de prédicat d'arité n
- ensemble $\mathcal{L}_0(\mathcal{F}, \mathcal{P})$ des **formules atomiques sans variable**
  - les symboles de $\mathcal{P}_0$ sont des formules atomiques de $\mathcal{L}_0(\mathcal{F}, \mathcal{P})$
  - si p est un symbole de prédicat d'arité n ($p \in \mathcal{P}_n$) et si $t_1, \cdots, t_n$ sont des termes dans $\mathcal{T}_0(\mathcal{F})$ alors $p(t_1, \cdots, t_n)$ est une formule atomique de $\mathcal{L}_0(\mathcal{F}, \mathcal{P})$

Connecteurs logiques

- Connecteur : construction d'une formule à partir d'autres formules

Connecteurs logiques

- Connecteur : construction d'une formule à partir d'autres formules
  - $\neg$: négation

Connecteurs logiques

- Connecteur : construction d'une formule à partir d'autres formules
  - $\neg$: négation
  - $\Rightarrow$: implication

Connecteurs logiques

- Connecteur : construction d'une formule à partir d'autres formules
  - $\neg$: négation
  - $\Rightarrow$: implication
  - $\wedge$: conjonction (et)

Connecteurs logiques

- Connecteur : construction d'une formule à partir d'autres formules
  - $\neg$: négation
  - $\Rightarrow$: implication
  - $\wedge$: conjonction (et)
  - $\vee$: disjonction (ou)

Connecteurs logiques

- Connecteur : construction d'une formule à partir d'autres formules
  - $\neg$: négation
  - $\Rightarrow$: implication
  - $\wedge$: conjonction (et)
  - $\vee$: disjonction (ou)
- *exemple :* « *Si Socrate ne lit pas ses messages sur son smartphone et qu'il suit un cours de logique, alors il comprend la logique.* »

Connecteurs logiques

- Connecteur : construction d'une formule à partir d'autres formules
  - $\neg$: négation
  - $\Rightarrow$: implication
  - $\wedge$: conjonction (et)
  - $\vee$: disjonction (ou)
- *exemple :* « *Si Socrate ne lit pas ses messages sur son smartphone et qu'il suit un cours de logique, alors il comprend la logique.* »

formule	énoncé
F_1	*Socrate suit un cours de logique*
F_2	*Socrate lit ses messages sur son smartphone*
F_3	*Socrate comprend la logique*

Connecteurs logiques

- Connecteur : construction d'une formule à partir d'autres formules
  - $\neg$: négation
  - $\Rightarrow$: implication
  - $\wedge$: conjonction (et)
  - $\vee$: disjonction (ou)
- *exemple :* « *Si Socrate ne lit pas ses messages sur son smartphone et qu'il suit un cours de logique, alors il comprend la logique.* »

formule	énoncé
F_1	*Socrate suit un cours de logique*
F_2	*Socrate lit ses messages sur son smartphone*
F_3	*Socrate comprend la logique*

$$(\neg F_2 \wedge F_1) \Rightarrow F_3$$

Connecteurs logiques

- Connecteur : construction d'une formule à partir d'autres formules
  - $\neg$: négation
  - $\Rightarrow$: implication
  - $\wedge$: conjonction (et)
  - $\vee$: disjonction (ou)
- *exemple :* « *Si il ne pleut pas et que je suis en vacances, alors je vais à la plage ou je fais de la logique.* »

Connecteurs logiques

- Connecteur : construction d'une formule à partir d'autres formules
  - $\neg$: négation
  - $\Rightarrow$: implication
  - $\wedge$: conjonction (et)
  - $\vee$: disjonction (ou)
- *exemple :* « *Si il ne pleut pas et que je suis en vacances, alors je vais à la plage ou je fais de la logique.* »

$$(\neg F_1 \wedge F_2) \Rightarrow (F_3 \vee F_4)$$

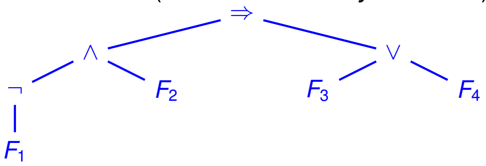
F_1	*il pleut*	F_2	*je suis en vacances*
F_3	*je vais à la plage*	F_4	*je fais de la logique*

Connecteurs logiques

- Connecteur : construction d'une formule à partir d'autres formules
  - $\neg$: négation
  - $\Rightarrow$: implication
  - $\wedge$: conjonction (et)
  - $\vee$: disjonction (ou)
- *exemple :* « *Si il ne pleut pas et que je suis en vacances, alors je vais à la plage ou je fais de la logique.* »

$$(\neg F_1 \wedge F_2) \Rightarrow (F_3 \vee F_4)$$

arbre de syntaxe abstraite (AST : *Abstract Syntax Tree*)



Connecteurs logiques

- Connecteur : construction d'une formule à partir d'autres formules
  - $\neg$: négation
  - $\Rightarrow$: implication
  - $\wedge$: conjonction (et)
  - $\vee$: disjonction (ou)
- raccourci : on écrit $F_1 \Leftrightarrow F_2$ pour exprimer $(F_1 \Rightarrow F_2) \wedge (F_2 \Rightarrow F_1)$

Langage des formules logiques

- **définition inductive de l'ensemble $\mathbb{F}_0(\mathcal{F}, \mathcal{P})$ des formules logiques sans variable**

Langage des formules logiques

- **définition inductive de l'ensemble $\mathbb{F}_0(\mathcal{F}, \mathcal{P})$ des formules logiques sans variable**
  - toute formule atomique de $\mathcal{L}_0(\mathcal{F}, \mathcal{P})$ est une formule logique

Langage des formules logiques

- **définition inductive de l'ensemble $\mathbb{F}_0(\mathcal{F}, \mathcal{P})$ des formules logiques sans variable**
  - toute formule atomique de $\mathcal{L}_0(\mathcal{F}, \mathcal{P})$ est une formule logique
  - true et false sont des formules logiques

Langage des formules logiques

- **définition inductive de l'ensemble $\mathbb{F}_0(\mathcal{F}, \mathcal{P})$ des formules logiques sans variable**
  - toute formule atomique de $\mathcal{L}_0(\mathcal{F}, \mathcal{P})$ est une formule logique
  - true et false sont des formules logiques
  - si F est une formule logique, alors $\neg F$ est aussi une formule logique

Langage des formules logiques

- **définition inductive de l'ensemble $\mathbb{F}_0(\mathcal{F}, \mathcal{P})$ des formules logiques sans variable**
  - toute formule atomique de $\mathcal{L}_0(\mathcal{F}, \mathcal{P})$ est une formule logique
  - true et false sont des formules logiques
  - si F est une formule logique, alors $\neg F$ est aussi une formule logique
  - si F_1 et F_2 sont des formules logiques, alors $(F_1 \wedge F_2)$, $(F_1 \vee F_2)$ et $(F_1 \Rightarrow F_2)$ sont aussi des formules logiques

Langage des formules logiques

- **définition inductive de l'ensemble $\mathbb{F}_0(\mathcal{F}, \mathcal{P})$ des formules logiques sans variable**
  - toute formule atomique de $\mathcal{L}_0(\mathcal{F}, \mathcal{P})$ est une formule logique
  - true et false sont des formules logiques
  - si F est une formule logique, alors $\neg F$ est aussi une formule logique
  - si F_1 et F_2 sont des formules logiques, alors $(F_1 \wedge F_2)$, $(F_1 \vee F_2)$ et $(F_1 \Rightarrow F_2)$ sont aussi des formules logiques

  toutes les formules logiques sont obtenues en appliquant un nombre fini de fois ces règles de construction

Langage des formules logiques

- **définition inductive de l'ensemble $\mathbb{F}_0(\mathcal{F}, \mathcal{P})$ des formules logiques sans variable**
  - toute formule atomique de $\mathcal{L}_0(\mathcal{F}, \mathcal{P})$ est une formule logique
  - true et false sont des formules logiques
  - si F est une formule logique, alors $\neg F$ est aussi une formule logique
  - si F_1 et F_2 sont des formules logiques, alors $(F_1 \wedge F_2)$, $(F_1 \vee F_2)$ et $(F_1 \Rightarrow F_2)$ sont aussi des formules logiques
- on omet souvent certaines parenthèses « inutiles »

Langage des formules logiques

- **définition inductive de l'ensemble $\mathbb{F}_0(\mathcal{F}, \mathcal{P})$ des formules logiques sans variable**
  - toute formule atomique de $\mathcal{L}_0(\mathcal{F}, \mathcal{P})$ est une formule logique
  - true et false sont des formules logiques
  - si F est une formule logique, alors $\neg F$ est aussi une formule logique
  - si F_1 et F_2 sont des formules logiques, alors $(F_1 \wedge F_2)$, $(F_1 \vee F_2)$ et $(F_1 \Rightarrow F_2)$ sont aussi des formules logiques
- on omet souvent certaines parenthèses « inutiles »
  - **syntaxe concrète** : on utilise des parenthèses pour indiquer sur quelles formules portent les connecteurs logiques

Langage des formules logiques

- **définition inductive de l'ensemble $\mathbb{F}_0(\mathcal{F}, \mathcal{P})$ des formules logiques sans variable**
  - toute formule atomique de $\mathcal{L}_0(\mathcal{F}, \mathcal{P})$ est une formule logique
  - true et false sont des formules logiques
  - si F est une formule logique, alors $\neg F$ est aussi une formule logique
  - si F_1 et F_2 sont des formules logiques, alors $(F_1 \wedge F_2)$, $(F_1 \vee F_2)$ et $(F_1 \Rightarrow F_2)$ sont aussi des formules logiques
- on omet souvent certaines parenthèses « inutiles »
  - **syntaxe concrète** : on utilise des parenthèses pour indiquer sur quelles formules portent les connecteurs logiques
  - arbre de **syntaxe abstraite** : les formules sont représentées par des arbres et ne contiennent pas de parenthèses

Quantificateurs

- *exemple :*

$$t_4 = \begin{array}{|c|c|c|c|} \multicolumn{1}{c}{0} & \multicolumn{1}{c}{1} & \multicolumn{1}{c}{2} & \multicolumn{1}{c}{3} \\ \hline pos(t_4, 0) & pos(t_4, 1) & pos(t_4, 2) & pos(t_4, 3) \\ \hline \end{array}$$

Quantificateurs

- *exemple :*

$$t_4 = \begin{array}{cccc} 0 & 1 & 2 & 3 \\ \hline \multicolumn{1}{|c|}{pos(t_4, 0)} & \multicolumn{1}{c|}{pos(t_4, 1)} & \multicolumn{1}{c|}{pos(t_4, 2)} & \multicolumn{1}{c|}{pos(t_4, 3)} \\ \hline \end{array}$$

« **un** *tableau t_4 contenant 4 entiers est trié par ordre croissant* »

Quantificateurs

- *exemple :*

$$t_4 = \begin{array}{|c|c|c|c|} \multicolumn{1}{c}{0} & \multicolumn{1}{c}{1} & \multicolumn{1}{c}{2} & \multicolumn{1}{c}{3} \\ \hline pos(t_4, 0) & pos(t_4, 1) & pos(t_4, 2) & pos(t_4, 3) \\ \hline \end{array}$$

« <u>**un**</u> *tableau t_4 contenant 4 entiers est trié par ordre croissant* »

$\leq (pos(t_4, 0), pos(t_4, 1))$ et $\leq (pos(t_4, 1), pos(t_4, 2))$ et $\leq (pos(t_4, 2), pos(t_4, 3))$

Quantificateurs

- *exemple :*

  « <u>**un**</u> *tableau* t_{99} *contenant 99 entiers est trié par ordre croissant* »

Quantificateurs

- *exemple :*

  « **un** *tableau* t_{99} *contenant 99 entiers est trié par ordre croissant* »

  - $\leq(pos(t_{99}, 0), pos(t_{99}, 1)) \wedge \cdots \wedge \leq(pos(t_{99}, 97), pos(t_{99}, 98))$

Quantificateurs

- *exemple :*

  « **un** *tableau t_{99} contenant 99 entiers est trié par ordre croissant* »

  - $\leq(pos(t_{99}, 0), pos(t_{99}, 1)) \wedge \cdots \wedge \leq(pos(t_{99}, 97), pos(t_{99}, 98))$
  - $\bigwedge_{i=0}^{i=97} \leq(pos(t_{99}, \boxed{i}), pos(t_{99}, \boxed{i+1}))$

Quantificateurs

- *exemple :*

  « **un** *tableau t_{99} contenant 99 entiers est trié par ordre croissant* »

  - $\leq(pos(t_{99},0), pos(t_{99},1)) \wedge \cdots \wedge \leq(pos(t_{99},97), pos(t_{99},98))$
  - $\bigwedge_{i=0}^{i=97} \leq(pos(t_{99},\boxed{i}), pos(t_{99},\boxed{i+1}))$
  - **pour tout** i,

    si $\leq(0,\boxed{i})$ et $\leq(\boxed{i},97)$, alors $\leq(pos(t_{99},\boxed{i}), pos(t_{99},\boxed{i+1}))$

Quantificateurs

- *exemple :*

  « **un** *tableau* t_{99} *contenant 99 entiers est trié par ordre croissant* »

  - $\leq (pos(t_{99}, 0), pos(t_{99}, 1)) \wedge \cdots \wedge \leq (pos(t_{99}, 97), pos(t_{99}, 98))$
  - $\bigwedge_{i=0}^{i=97} \leq (pos(t_{99}, \boxed{i}), pos(t_{99}, \boxed{i+1}))$
  - **pour tout** i,

    si $\leq (0, \boxed{i})$ et $\leq (\boxed{i}, 97)$, alors $\leq (pos(t_{99}, \boxed{i}), pos(t_{99}, \boxed{i+1}))$

pour tous les tableaux d'entiers ?

Quantificateurs

- *exemple :*

  « **un** *tableau* t_{99} *contenant 99 entiers est trié par ordre croissant* »

  - $\leq (pos(t_{99}, 0), pos(t_{99}, 1)) \wedge \cdots \wedge \leq (pos(t_{99}, 97), pos(t_{99}, 98))$
  - $\bigwedge_{i=0}^{i=97} \leq (pos(t_{99}, \boxed{i}), pos(t_{99}, \boxed{i+1}))$
  - **pour tout** i,

    si $\leq (0, \boxed{i})$ et $\leq (\boxed{i}, 97)$, alors $\leq (pos(t_{99}, \boxed{i}), pos(t_{99}, \boxed{i+1}))$

pour tous les tableaux d'entiers ?

abstraction : ne pas dépendre du nombre d'éléments du tableau

Quantificateurs

- *exemple :*
  « **un** *tableau d'entiers* t *est trié par ordre croissant* »

Quantificateurs

- *exemple :*

  « **un** *tableau d'entiers t est trié par ordre croissant* »

  - $len(t)$: nombre d'éléments du tableau t ($len \in \mathcal{F}_1$)

Quantificateurs

- *exemple :*

  « **un** *tableau d'entiers t est trié par ordre croissant* »

  - $len(t)$: nombre d'éléments du tableau t ($len \in \mathcal{F}_1$)
  - pour tout t $\quad$ est_trié(t)

    si et seulement si

    $$\left(\begin{array}{l} \text{pour tout } i \\ \text{si } \leq(0, i) \text{ et } \leq(i, -(len(t), 2)) \\ \text{alors } \leq(pos(t, i), pos(t, +(i, 1))) \end{array} \right)$$

Quantificateurs

- *exemple :*

  « **un** *tableau d'entiers t est trié par ordre croissant* »

  - $len(t)$: nombre d'éléments du tableau t ($len \in \mathcal{F}_1$)

    pour tout t est_trié(t)
    si et seulement si

  - $$\left(\begin{array}{l} \text{pour tout } i \\ \text{si } \leq(0,i) \text{ et } \leq(i,-(len(t),2)) \\ \text{alors } \leq(pos(t,i),pos(t,+(i,1))) \end{array} \right)$$

  - $$\forall t \quad \text{est\_trié}(t) \Leftrightarrow \forall i \left(\begin{array}{ll} & (\leq(0,i) \wedge \leq(i,-(len(t),2))) \\ \Rightarrow & \leq(pos(t,i),pos(t,+(i,1))) \end{array} \right)$$

Quantificateurs

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)

- *exemple :*

  « **un** *tableau d'entiers t est trié par ordre croissant* »

  - $len(t)$: nombre d'éléments du tableau t ($len \in \mathcal{F}_1$)

    pour tout t est_trié(t)
    si et seulement si
  - $$\left(\begin{array}{l} \text{pour tout } i \\ \text{si } \le(0,i) \text{ et } \le(i, -(len(t),2)) \\ \text{alors } \le(pos(t,i), pos(t,+(i,1))) \end{array}\right)$$
  - $$\forall t \quad \text{est\_trié}(t) \Leftrightarrow \forall i \left(\begin{array}{ll} & (\le(0,i) \wedge \le(i,-(len(t),2))) \\ \Rightarrow & \le(pos(t,i), pos(t,+(i,1))) \end{array}\right)$$
  - i et t sont des **variables** *quantifiées* (universellement $\forall$)

Quantificateurs

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)

- *exemple :* « *la relation binaire r est réflexive* »

Quantificateurs

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)
- *exemple :* « *la relation binaire r est réflexive* »
  - $\forall x \; r(x, x)$

Quantificateurs

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)

- *exemple :* « *la relation binaire r est réflexive* »
  - $\forall x \; r(x, x)$
  - quantification universelle devant une formule atomique

Quantificateurs

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)
- *exemple :* « *quelqu'un suit un cours de logique* »

Quantificateurs

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)
  - quantificateur existentiel : $\exists$ (il-existe)
- *exemple :* « *quelqu'un suit un cours de logique* »
  - $\exists x \ suit\_un\_cours\_de\_logique(x)$

Quantificateurs

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)
  - quantificateur existentiel : $\exists$ (il-existe)
- *exemple :* « *quelqu'un suit un cours de logique* »
  - $\exists x\ suit\_un\_cours\_de\_logique(x)$
  - quantification existentielle devant une formule atomique

Quantificateurs

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)
  - quantificateur existentiel : $\exists$ (il-existe)
- *exemple :* « *tous les étudiants ont un idéal* »

Quantificateurs

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)
  - quantificateur existentiel : $\exists$ (il-existe)
- *exemple :* « *tous les étudiants ont un idéal* »
  - $\forall x\,(etudiant(x) \Rightarrow \exists y\,(ideal(y) \wedge ideal\_of(y, x)))$

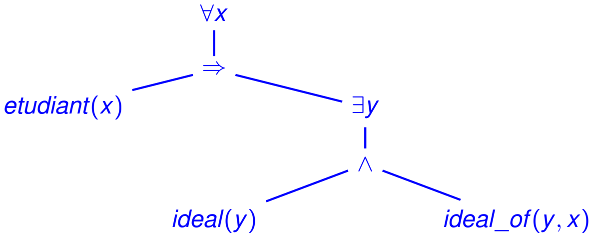
Quantificateurs

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)
  - quantificateur existentiel : $\exists$ (il-existe)
- *exemple :* « *tous les étudiants ont un idéal* »
  - $\forall x\,(etudiant(x) \Rightarrow \exists y\,(ideal(y) \wedge ideal\_of(y, x)))$
  - quantifications devant des formules logiques

Quantificateurs

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)
  - quantificateur existentiel : $\exists$ (il-existe)
- *exemple :* « *tous les étudiants ont un idéal* »
  - $\forall x\,(\mathit{etudiant}(x) \Rightarrow \exists y\,(\mathit{ideal}(y) \wedge \mathit{ideal\_of}(y, x)))$
  - quantifications devant des formules logiques

arbre de syntaxe abstraite (AST : *Abstract Syntax Tree*)



Quantificateurs

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)
  - quantificateur existentiel : $\exists$ (il-existe)
- *exemple :* « *pour tout x qui vérifie $p(x)$, il existe y tel que $q(x, y)$* »

Quantificateurs

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)
  - quantificateur existentiel : $\exists$ (il-existe)
- *exemple :* « *pour tout x qui vérifie $p(x)$, il existe y tel que $q(x,y)$* »
  - $\forall x\,(p(x) \Rightarrow \exists y\, q(x,y))$

Quantificateurs

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)
  - quantificateur existentiel : $\exists$ (il-existe)
- *exemple :* « *pour tout x qui vérifie $p(x)$, il existe y tel que $q(x,y)$* »
  - $\forall x\,(p(x) \Rightarrow \exists y\, q(x,y))$
  - spécification de programmes : le programme f prend un argument x qui vérifie l'hypothèse $p(x)$ et retourne un résultat y (qui dépend de l'argument x) tel que $q(x,y)$:

$$\forall x\ (p(x) \Rightarrow q(x, f(x)))$$

Quantificateurs

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)
  - quantificateur existentiel : $\exists$ (il-existe)
- ambiguïtés : « *tout le monde aime quelqu'un* »

Quantificateurs

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)
  - quantificateur existentiel : $\exists$ (il-existe)
- ambiguïtés : « *tout le monde aime quelqu'un* »
  - il existe quelqu'un que tout le monde aime ?

Quantificateurs

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)
  - quantificateur existentiel : $\exists$ (il-existe)
- ambiguïtés : « *tout le monde aime quelqu'un* »
  - il existe quelqu'un que tout le monde aime ?

$$\exists x \ \forall y \ aime(y, x)$$

Quantificateurs

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)
  - quantificateur existentiel : $\exists$ (il-existe)
- ambiguïtés : « *tout le monde aime quelqu'un* »
  - il existe quelqu'un que tout le monde aime ?

$$\exists x \ \forall y \ aime(y, x)$$

  - pour chaque personne il existe quelqu'un aimé de cette personne ?

Quantificateurs

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)
  - quantificateur existentiel : $\exists$ (il-existe)
- ambiguïtés : « *tout le monde aime quelqu'un* »
  - il existe quelqu'un que tout le monde aime ?

$$\exists x \ \forall y \ \ aime(y, x)$$

  - pour chaque personne il existe quelqu'un aimé de cette personne ?

$$\forall y \ \exists x \ \ aime(y, x)$$

Quantificateurs

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)
  - quantificateur existentiel : $\exists$ (il-existe)
- ambiguïtés
  - « *Dans un triangle isocèle une médiane est également hauteur.* »

Quantificateurs

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)
  - quantificateur existentiel : $\exists$ (il-existe)
- ambiguïtés
  - « *Dans un triangle isocèle une médiane est également hauteur.* »

$$\forall x\ ((\mathrm{triangle}(x) \wedge \mathrm{isocele}(x)) \Rightarrow \exists y\ (\mathrm{mediane\_of}(y, x) \wedge \mathrm{hauteur\_of}(y, x)))$$

Quantificateurs

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)
  - quantificateur existentiel : $\exists$ (il-existe)
- ambiguïtés
  - « *Dans un triangle isocèle une médiane est également hauteur.* »

$$\forall x\ ((\mathrm{triangle}(x) \wedge \mathrm{isocele}(x)) \Rightarrow \exists y\ (\mathrm{mediane\_of}(y,x) \wedge \mathrm{hauteur\_of}(y,x)))$$

  - « *Dans un triangle équilatéral une médiane est également hauteur.* »

Quantificateurs

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)
  - quantificateur existentiel : $\exists$ (il-existe)
- ambiguïtés
  - « *Dans un triangle isocèle une médiane est également hauteur.* »

$$\forall x\ ((\mathrm{triangle}(x) \land \mathrm{isocele}(x)) \Rightarrow \exists y\ (\mathrm{mediane\_of}(y,x) \land \mathrm{hauteur\_of}(y,x)))$$

  - « *Dans un triangle équilatéral une médiane est également hauteur.* »

$$\forall x\ ((\mathrm{triangle}(x) \land \mathrm{equilateral}(x)) \Rightarrow \forall y\ (\mathrm{mediane\_of}(y,x) \Rightarrow \mathrm{hauteur\_of}(y,x)))$$

Quantificateurs

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)
  - quantificateur existentiel : $\exists$ (il-existe)
- raccourcis : quantification « bornée »
  - $\forall x \in E \ p(x)$

Quantificateurs

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)
  - quantificateur existentiel : $\exists$ (il-existe)
- raccourcis : quantification « bornée »
  - $\forall x \in E\ p(x)$ s'écrit : $\forall x\ (x \in E \Rightarrow p(x))$

    $\in$: prédicat d'arité 2

Quantificateurs

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)
  - quantificateur existentiel : $\exists$ (il-existe)
- raccourcis : quantification « bornée »
  - $\forall x \in E \; p(x)$ s'écrit : $\forall x \; (x \in E \Rightarrow p(x))$

    $\in$: prédicat d'arité 2
  - $\forall x > 5 \; p(x)$

Quantificateurs

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)
  - quantificateur existentiel : $\exists$ (il-existe)
- raccourcis : quantification « bornée »
  - $\forall x \in E \; p(x)$ s'écrit : $\forall x \; (x \in E \Rightarrow p(x))$
    $\in$: prédicat d'arité 2
  - $\forall x > 5 \; p(x)$ s'écrit : $\forall x \; (x > 5 \Rightarrow p(x))$
    $>$: prédicat d'arité 2

Quantificateurs

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)
  - quantificateur existentiel : $\exists$ (il-existe)
- raccourcis : quantification « bornée »
  - $\forall x \in E\ p(x)$ s'écrit : $\forall x\ (x \in E \Rightarrow p(x))$

    $\in$: prédicat d'arité 2
  - $\forall x > 5\ p(x)$ s'écrit : $\forall x\ (x > 5 \Rightarrow p(x))$

    $>$: prédicat d'arité 2
  - $\exists x \in E\ p(x)$

Quantificateurs

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)
  - quantificateur existentiel : $\exists$ (il-existe)
- raccourcis : quantification « bornée »
  - $\forall x \in E\ p(x)$ s'écrit : $\forall x\ (x \in E \Rightarrow p(x))$

    $\in$: prédicat d'arité 2
  - $\forall x > 5\ p(x)$ s'écrit : $\forall x\ (x > 5 \Rightarrow p(x))$

    $>$: prédicat d'arité 2
  - $\exists x \in E\ p(x)$ s'écrit : $\exists x\ (x \in E \wedge p(x))$

Quantificateurs

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)
  - quantificateur existentiel : $\exists$ (il-existe)
- raccourcis : quantification « bornée »
  - $\forall x \in E\ p(x)$ s'écrit : $\forall x\ (x \in E \Rightarrow p(x))$

    $\in$: prédicat d'arité 2
  - $\forall x > 5\ p(x)$ s'écrit : $\forall x\ (x > 5 \Rightarrow p(x))$

    $>$: prédicat d'arité 2
  - $\exists x \in E\ p(x)$ s'écrit : $\exists x\ (x \in E \wedge p(x))$
  - $\exists x > 5\ p(x)$

Quantificateurs

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)
  - quantificateur existentiel : $\exists$ (il-existe)
- raccourcis : quantification « bornée »
  - $\forall x \in E\ p(x)$ s'écrit : $\forall x\ (x \in E \Rightarrow p(x))$

    $\in$: prédicat d'arité 2
  - $\forall x > 5\ p(x)$ s'écrit : $\forall x\ (x > 5 \Rightarrow p(x))$

    $>$: prédicat d'arité 2
  - $\exists x \in E\ p(x)$ s'écrit : $\exists x\ (x \in E \wedge p(x))$
  - $\exists x > 5\ p(x)$ s'écrit : $\exists x\ (x > 5 \wedge p(x))$

Quantificateurs du 1er ordre

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)
  - quantificateur existentiel : $\exists$ (il-existe)
- les énoncés peuvent exprimer qu'une propriété est vérifiée pour tous les objets ou seulement qu'il existe des objets pour lesquels cette propriété est vérifiée

Quantificateurs du 1er ordre

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)
  - quantificateur existentiel : $\exists$ (il-existe)
- les énoncés peuvent exprimer qu'une propriété est vérifiée pour tous les objets ou seulement qu'il existe des objets pour lesquels cette propriété est vérifiée
  - 1er ordre : quantification des symboles de variable désignant des objets sur lesquels l'énoncé porte

Quantificateurs du 1er ordre

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)
  - quantificateur existentiel : $\exists$ (il-existe)
- les énoncés peuvent exprimer qu'une propriété est vérifiée pour tous les objets ou seulement qu'il existe des objets pour lesquels cette propriété est vérifiée
  - 1er ordre : quantification des symboles de variable désignant des objets sur lesquels l'énoncé porte
  - on ne quantifie pas sur les symboles de fonction :

$$\forall f\ (\mathrm{injective}(f) \Leftrightarrow \forall x\, \forall y\, (f(x) = f(y) \Rightarrow x = y)) \notin \mathbb{F}(X, \mathcal{F}, \mathcal{P})$$

Quantificateurs du 1er ordre

- quantificateurs
  - quantificateur universel : $\forall$ (pour-tout)
  - quantificateur existentiel : $\exists$ (il-existe)
- les énoncés peuvent exprimer qu'une propriété est vérifiée pour tous les objets ou seulement qu'il existe des objets pour lesquels cette propriété est vérifiée
  - 1er ordre : quantification des symboles de variable désignant des objets sur lesquels l'énoncé porte
  - on ne quantifie pas sur les symboles de fonction :

$$\forall f\ (\mathrm{injective}(f) \Leftrightarrow \forall x\, \forall y\, (f(x) = f(y) \Rightarrow x = y)) \notin \mathbb{F}(X, \mathcal{F}, \mathcal{P})$$

ni sur les symboles de prédicat :

$$\forall p\ (p(0) \wedge (\forall n\, p(n) \Rightarrow p(n+1))) \Rightarrow \forall n\, p(n) \notin \mathbb{F}(X, \mathcal{F}, \mathcal{P})$$

($\neq$ logiques d'ordre supérieur)

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : **les termes**

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : **les termes**

- les termes peuvent être :
  - des symboles de **variables** x, y, z, etc.

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : **les termes**

- les termes peuvent être :
  - des symboles de **variables** x, y, z, etc.
  - des noms d'objets/d'individus Socrate, 2, 3, 5 : **les constantes**

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : **les termes**

- les termes peuvent être :
  - des symboles de **variables** x, y, z, etc.
  - des noms d'objets/d'individus Socrate, 2, 3, 5 : **les constantes**
  - des **fonctions** appliquées à des objets $+(7, y)$ ou plus généralement des fonctions appliquées à d'autres termes $+(+(x, 3), 4)$

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : **les termes**

- les termes peuvent être :
  - des symboles de **variables** x, y, z, etc.
  - des noms d'objets/d'individus Socrate, 2, 3, 5 : **les constantes**
  - des **fonctions** appliquées à des objets $+(7, y)$ ou plus généralement des fonctions appliquées à d'autres termes $+(+(x, 3), 4)$

- définition inductive des termes

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : **les termes**

  X : ensemble de symboles de variable
  $\mathcal{F}$: ensemble de

  symboles de constante
  symboles de fonction

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : **les termes**

  X : ensemble de symboles de variable
  $\mathcal{F}$: ensemble de
    symboles de constante
    symboles de fonction

- *exemple :* expressions arithmétiques (simples)

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : **les termes**

  $X = \{x, y, z, \cdots\}$: ensemble de symboles de variable
  $\mathcal{F}$: ensemble de

  symboles de constante
  symboles de fonction

- *exemple :* expressions arithmétiques (simples)
  - les variables sont des expressions arithmétiques

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : **les termes**

  $X = \{x, y, z, \cdots\}$: ensemble de symboles de variable
  $\mathcal{F} = \mathcal{F}_0$: ensemble de symboles de constante et de fonction

  $\mathcal{F}_0 = \mathbb{Z}$: symboles de constante
  symboles de fonction

- *exemple :* expressions arithmétiques (simples)
  - les variables sont des expressions arithmétiques
  - les entiers relatifs sont des expressions arithmétiques

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : **les termes**

  $X = \{x, y, z, \cdots\}$: ensemble de symboles de variable
  $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_2$: ensemble de symboles de constante et de fonction

  $\mathcal{F}_0 = \mathbb{Z}$ contient les symboles de constante
  $\mathcal{F}_2 = \{+, -, \times, /\}$ contient les symboles de fonction d'arité 2

- *exemple :* expressions arithmétiques (simples)
  - les variables sont des expressions arithmétiques
  - les entiers relatifs sont des expressions arithmétiques
  - si e_1 et e_2 sont des expressions arithmétiques, alors $e_1 + e_2$, $e_1 - e_2$, $e_1 \times e_2$ et e_1 / e_2 sont des expressions arithmétiques

Langage de termes

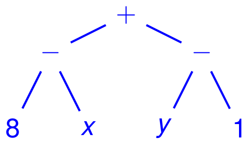
- les énoncés expriment des propriétés sur les objets de l'univers du discours : **les termes**

  $X = \{x, y, z, \cdots\}$: ensemble de symboles de variable
  $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_2$: ensemble de symboles de constante et de fonction

    $\mathcal{F}_0 = \mathbb{Z}$ contient les symboles de constante
    $\mathcal{F}_2 = \{+, -, \times, /\}$ contient les symboles de fonction d'arité 2

- *exemple :* expressions arithmétiques (simples)

  $(8 - x) + (y - 1)$



– arbre dont chaque nœud a 0 ou 2 fils
– feuilles de l'arbre : constantes dans $\mathbb{Z}$ et variables dans X
– chaque nœud étiqueté par $+$, $-$, $\times$ et $/$ a exactement 2 fils

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : **les termes**
  X : ensemble de symboles de variable
  $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_1 \cup \mathcal{F}_2 \cup \mathcal{F}_3 \cup \cdots = \bigcup_{n \geq 0} \mathcal{F}_n$: ensemble de symboles
    $\mathcal{F}_0$ contient les symboles de constante
    $\mathcal{F}_n$ contient les symboles de fonction d'arité n
- **définition inductive de l'ensemble $\mathcal{T}(X, \mathcal{F})$ des termes avec variable**

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : **les termes**
  X : ensemble de symboles de variable
  $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_1 \cup \mathcal{F}_2 \cup \mathcal{F}_3 \cup \cdots = \bigcup_{n \geq 0} \mathcal{F}_n$: ensemble de symboles
    $\mathcal{F}_0$ contient les symboles de constante
    $\mathcal{F}_n$ contient les symboles de fonction d'arité n
- **définition inductive de l'ensemble $\mathcal{T}(X, \mathcal{F})$ des termes avec variable**
  - toute variable $x \in X$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - toute constante $k \in \mathcal{F}_0$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - si $f \in \mathcal{F}_n$ et si $t_1, \cdots, t_n$ sont des termes dans $\mathcal{T}(X, \mathcal{F})$, alors $f(t_1, \cdots, t_n)$ est un terme dans $\mathcal{T}(X, \mathcal{F})$

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : **les termes**
  X : ensemble de symboles de variable
  $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_1 \cup \mathcal{F}_2 \cup \mathcal{F}_3 \cup \cdots = \bigcup_{n \geq 0} \mathcal{F}_n$: ensemble de symboles
- **définition inductive de l'ensemble $\mathcal{T}(X, \mathcal{F})$ des termes avec variable**
  - toute variable $x \in X$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - toute constante $k \in \mathcal{F}_0$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - si $f \in \mathcal{F}_n$ et si $t_1, \cdots, t_n$ sont des termes dans $\mathcal{T}(X, \mathcal{F})$, alors $f(t_1, \cdots, t_n)$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
- un terme $t \in \mathcal{T}(X, \mathcal{F})$ peut être vu comme un **arbre**

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : **les termes**
  X : ensemble de symboles de variable
  $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_1 \cup \mathcal{F}_2 \cup \mathcal{F}_3 \cup \cdots = \bigcup_{n \geq 0} \mathcal{F}_n$: ensemble de symboles
- **définition inductive de l'ensemble $\mathcal{T}(X, \mathcal{F})$ des termes avec variable**
  - toute variable $x \in X$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - toute constante $k \in \mathcal{F}_0$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - si $f \in \mathcal{F}_n$ et si $t_1, \cdots, t_n$ sont des termes dans $\mathcal{T}(X, \mathcal{F})$, alors $f(t_1, \cdots, t_n)$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
- un terme $t \in \mathcal{T}(X, \mathcal{F})$ peut être vu comme un **arbre**
  - les feuilles sont étiquetées par des éléments de X ou $\mathcal{F}_0$

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : **les termes**

  X : ensemble de symboles de variable

  $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_1 \cup \mathcal{F}_2 \cup \mathcal{F}_3 \cup \cdots = \bigcup_{n \geq 0} \mathcal{F}_n$: ensemble de symboles
- **définition inductive de l'ensemble $\mathcal{T}(X, \mathcal{F})$ des termes avec variable**
  - toute variable $x \in X$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - toute constante $k \in \mathcal{F}_0$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - si $f \in \mathcal{F}_n$ et si $t_1, \cdots, t_n$ sont des termes dans $\mathcal{T}(X, \mathcal{F})$, alors $f(t_1, \cdots, t_n)$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
- un terme $t \in \mathcal{T}(X, \mathcal{F})$ peut être vu comme un **arbre**
  - les feuilles sont étiquetées par des éléments de X ou $\mathcal{F}_0$
  - chaque nœud étiqueté par un élément de $\mathcal{F}_n$ a exactement n fils

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : **les termes**
  X : ensemble de symboles de variable
  $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_1 \cup \mathcal{F}_2 \cup \mathcal{F}_3 \cup \cdots = \bigcup_{n \geq 0} \mathcal{F}_n$: ensemble de symboles
- **définition inductive de l'ensemble $\mathcal{T}(X, \mathcal{F})$ des termes avec variable**
  - toute variable $x \in X$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - toute constante $k \in \mathcal{F}_0$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - si $f \in \mathcal{F}_n$ et si $t_1, \cdots, t_n$ sont des termes dans $\mathcal{T}(X, \mathcal{F})$, alors $f(t_1, \cdots, t_n)$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
- **raisonnement par induction structurelle**

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : **les termes**

  X : ensemble de symboles de variable

  $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_1 \cup \mathcal{F}_2 \cup \mathcal{F}_3 \cup \cdots = \bigcup_{n \geq 0} \mathcal{F}_n$: ensemble de symboles
- **définition inductive de l'ensemble $\mathcal{T}(X, \mathcal{F})$ des termes avec variable**
  - toute variable $x \in X$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - toute constante $k \in \mathcal{F}_0$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - si $f \in \mathcal{F}_n$ et si $t_1, \cdots, t_n$ sont des termes dans $\mathcal{T}(X, \mathcal{F})$, alors $f(t_1, \cdots, t_n)$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
- **raisonnement par induction structurelle** : si
  - $P(x)$ est vrai pour toute variable $x \in X$

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : **les termes**

  X : ensemble de symboles de variable

  $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_1 \cup \mathcal{F}_2 \cup \mathcal{F}_3 \cup \cdots = \bigcup_{n \geq 0} \mathcal{F}_n$: ensemble de symboles
- **définition inductive de l'ensemble $\mathcal{T}(X, \mathcal{F})$ des termes avec variable**
  - toute variable $x \in X$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - toute constante $k \in \mathcal{F}_0$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - si $f \in \mathcal{F}_n$ et si $t_1, \cdots, t_n$ sont des termes dans $\mathcal{T}(X, \mathcal{F})$, alors $f(t_1, \cdots, t_n)$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
- **raisonnement par induction structurelle** : si
  - $P(x)$ est vrai pour toute variable $x \in X$
  - $P(k)$ est vrai pour toute constante $k \in \mathcal{F}_0$

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : **les termes**

  X : ensemble de symboles de variable

  $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_1 \cup \mathcal{F}_2 \cup \mathcal{F}_3 \cup \cdots = \bigcup_{n \geq 0} \mathcal{F}_n$: ensemble de symboles
- **définition inductive de l'ensemble $\mathcal{T}(X, \mathcal{F})$ des termes avec variable**
  - toute variable $x \in X$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - toute constante $k \in \mathcal{F}_0$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - si $f \in \mathcal{F}_n$ et si $t_1, \cdots, t_n$ sont des termes dans $\mathcal{T}(X, \mathcal{F})$, alors $f(t_1, \cdots, t_n)$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
- **raisonnement par induction structurelle** : si
  - $P(x)$ est vrai pour toute variable $x \in X$
  - $P(k)$ est vrai pour toute constante $k \in \mathcal{F}_0$
  - pour tout $f \in \mathcal{F}_n$ et tous termes $t_1, \cdots, t_n$, si $P(t_1), \cdots, P(t_n)$ sont vrais, alors $P(f(t_1, \cdots, t_n))$ est vrai

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : **les termes**

  X : ensemble de symboles de variable

  $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_1 \cup \mathcal{F}_2 \cup \mathcal{F}_3 \cup \cdots = \bigcup_{n \geq 0} \mathcal{F}_n$: ensemble de symboles
- **définition inductive de l'ensemble $\mathcal{T}(X, \mathcal{F})$ des termes avec variable**
  - toute variable $x \in X$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - toute constante $k \in \mathcal{F}_0$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - si $f \in \mathcal{F}_n$ et si $t_1, \cdots, t_n$ sont des termes dans $\mathcal{T}(X, \mathcal{F})$, alors $f(t_1, \cdots, t_n)$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
- **raisonnement par induction structurelle** : si
  - $P(x)$ est vrai pour toute variable $x \in X$
  - $P(k)$ est vrai pour toute constante $k \in \mathcal{F}_0$
  - pour tout $f \in \mathcal{F}_n$ et tous termes $t_1, \cdots, t_n$, si $P(t_1), \cdots, P(t_n)$ sont vrais, alors $P(f(t_1, \cdots, t_n))$ est vrai

  alors pour tout terme $t \in \mathcal{T}(X, \mathcal{F})$, $P(t)$ est vrai.

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : **les termes**
  X : ensemble de symboles de variable
  $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_1 \cup \mathcal{F}_2 \cup \mathcal{F}_3 \cup \cdots = \bigcup_{n \geq 0} \mathcal{F}_n$: ensemble de symboles
- **définition inductive de l'ensemble $\mathcal{T}(X, \mathcal{F})$ des termes avec variable**
  - toute variable $x \in X$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - toute constante $k \in \mathcal{F}_0$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - si $f \in \mathcal{F}_n$ et si $t_1, \cdots, t_n$ sont des termes dans $\mathcal{T}(X, \mathcal{F})$, alors $f(t_1, \cdots, t_n)$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
- **définition inductive (récursive) de fonction**

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : **les termes**
  X : ensemble de symboles de variable
  $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_1 \cup \mathcal{F}_2 \cup \mathcal{F}_3 \cup \cdots = \bigcup_{n \geq 0} \mathcal{F}_n$: ensemble de symboles
- **définition inductive de l'ensemble $\mathcal{T}(X, \mathcal{F})$ des termes avec variable**
  - toute variable $x \in X$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - toute constante $k \in \mathcal{F}_0$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - si $f \in \mathcal{F}_n$ et si $t_1, \cdots, t_n$ sont des termes dans $\mathcal{T}(X, \mathcal{F})$, alors $f(t_1, \cdots, t_n)$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
- **définition inductive (récursive) de fonction**
  - variables apparaissant dans un terme $\vartheta : \mathcal{T}(X, \mathcal{F}) \to \wp(X)$

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : **les termes**

  X : ensemble de symboles de variable

  $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_1 \cup \mathcal{F}_2 \cup \mathcal{F}_3 \cup \cdots = \bigcup_{n \geq 0} \mathcal{F}_n$: ensemble de symboles
- **définition inductive de l'ensemble $\mathcal{T}(X, \mathcal{F})$ des termes avec variable**
  - toute variable $x \in X$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - toute constante $k \in \mathcal{F}_0$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - si $f \in \mathcal{F}_n$ et si $t_1, \cdots, t_n$ sont des termes dans $\mathcal{T}(X, \mathcal{F})$, alors $f(t_1, \cdots, t_n)$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
- **définition inductive (récursive) de fonction**
  - variables apparaissant dans un terme $\vartheta : \mathcal{T}(X, \mathcal{F}) \to \wp(X)$

$$\vartheta(t) = \begin{cases} \{x\} & \text{si } t = x \in X \\ & \\ & \end{cases}$$

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : **les termes**
  X : ensemble de symboles de variable
  $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_1 \cup \mathcal{F}_2 \cup \mathcal{F}_3 \cup \cdots = \bigcup_{n \geq 0} \mathcal{F}_n$: ensemble de symboles
- **définition inductive de l'ensemble $\mathcal{T}(X, \mathcal{F})$ des termes avec variable**
  - toute variable $x \in X$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - toute constante $k \in \mathcal{F}_0$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - si $f \in \mathcal{F}_n$ et si $t_1, \cdots, t_n$ sont des termes dans $\mathcal{T}(X, \mathcal{F})$, alors $f(t_1, \cdots, t_n)$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
- **définition inductive (récursive) de fonction**
  - variables apparaissant dans un terme $\vartheta : \mathcal{T}(X, \mathcal{F}) \rightarrow \wp(X)$

$$\vartheta(t) = \begin{cases} \{x\} & \text{si } t = x \in X \\ \emptyset & \text{si } t = k \in \mathcal{F}_0 \\ & \end{cases}$$

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : **les termes**
  X : ensemble de symboles de variable
  $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_1 \cup \mathcal{F}_2 \cup \mathcal{F}_3 \cup \cdots = \bigcup_{n \geq 0} \mathcal{F}_n$: ensemble de symboles
- **définition inductive de l'ensemble $\mathcal{T}(X, \mathcal{F})$ des termes avec variable**
  - toute variable $x \in X$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - toute constante $k \in \mathcal{F}_0$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - si $f \in \mathcal{F}_n$ et si $t_1, \cdots, t_n$ sont des termes dans $\mathcal{T}(X, \mathcal{F})$, alors $f(t_1, \cdots, t_n)$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
- **définition inductive (récursive) de fonction**
  - variables apparaissant dans un terme $\vartheta : \mathcal{T}(X, \mathcal{F}) \to \wp(X)$

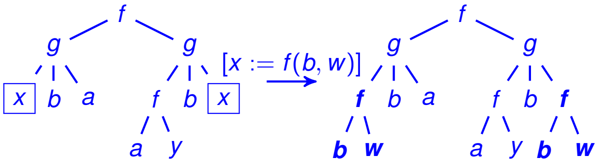
$$\vartheta(t) = \begin{cases} \{x\} & \text{si } t = x \in X \\ \emptyset & \text{si } t = k \in \mathcal{F}_0 \\ \vartheta(t_1) \cup \cdots \cup \vartheta(t_n) & \text{si } t = f(t_1, \cdots, t_n) \end{cases}$$

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : **les termes**

  X : ensemble de symboles de variable

  $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_1 \cup \mathcal{F}_2 \cup \mathcal{F}_3 \cup \cdots = \bigcup_{n \geq 0} \mathcal{F}_n$: ensemble de symboles
- **définition inductive de l'ensemble $\mathcal{T}(X, \mathcal{F})$ des termes avec variable**
  - toute variable $x \in X$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - toute constante $k \in \mathcal{F}_0$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - si $f \in \mathcal{F}_n$ et si $t_1, \cdots, t_n$ sont des termes dans $\mathcal{T}(X, \mathcal{F})$, alors $f(t_1, \cdots, t_n)$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
- **définition inductive (récursive) de fonction**
  - terme $t[x := t']$: remplacer x par t' dans t



Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : **les termes**
  X : ensemble de symboles de variable
  $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_1 \cup \mathcal{F}_2 \cup \mathcal{F}_3 \cup \cdots = \bigcup_{n \geq 0} \mathcal{F}_n$: ensemble de symboles
- **définition inductive de l'ensemble $\mathcal{T}(X, \mathcal{F})$ des termes avec variable**
  - toute variable $x \in X$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - toute constante $k \in \mathcal{F}_0$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - si $f \in \mathcal{F}_n$ et si $t_1, \cdots, t_n$ sont des termes dans $\mathcal{T}(X, \mathcal{F})$, alors $f(t_1, \cdots, t_n)$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
- **définition inductive (récursive) de fonction**
  - terme $t[x := t']$: remplacer x par t' dans t

$$t[x := t'] = \begin{cases} y & \text{si } t = y \in X \text{ et } x \neq y \\ & \\ & \end{cases}$$

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : **les termes**

  X : ensemble de symboles de variable

  $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_1 \cup \mathcal{F}_2 \cup \mathcal{F}_3 \cup \cdots = \bigcup_{n \geq 0} \mathcal{F}_n$: ensemble de symboles
- **définition inductive de l'ensemble $\mathcal{T}(X, \mathcal{F})$ des termes avec variable**
  - toute variable $x \in X$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - toute constante $k \in \mathcal{F}_0$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - si $f \in \mathcal{F}_n$ et si $t_1, \cdots, t_n$ sont des termes dans $\mathcal{T}(X, \mathcal{F})$, alors $f(t_1, \cdots, t_n)$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
- **définition inductive (récursive) de fonction**
  - terme $t[x := t']$: remplacer x par t' dans t

$$t[x := t'] = \begin{cases} y & \text{si } t = y \in X \text{ et } x \neq y \\ t' & \text{si } t = x \in X \\ & \\ & \end{cases}$$

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : **les termes**

  X : ensemble de symboles de variable

  $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_1 \cup \mathcal{F}_2 \cup \mathcal{F}_3 \cup \cdots = \bigcup_{n \geq 0} \mathcal{F}_n$: ensemble de symboles
- **définition inductive de l'ensemble $\mathcal{T}(X, \mathcal{F})$ des termes avec variable**
  - toute variable $x \in X$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - toute constante $k \in \mathcal{F}_0$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - si $f \in \mathcal{F}_n$ et si $t_1, \cdots, t_n$ sont des termes dans $\mathcal{T}(X, \mathcal{F})$, alors $f(t_1, \cdots, t_n)$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
- **définition inductive (récursive) de fonction**
  - terme $t[x := t']$: remplacer x par t' dans t

$$t[x := t'] = \begin{cases} y & \text{si } t = y \in X \text{ et } x \neq y \\ t' & \text{si } t = x \in X \\ k & \text{si } t = k \in \mathcal{F}_0 \\ & \end{cases}$$

Langage de termes

- les énoncés expriment des propriétés sur les objets de l'univers du discours : **les termes**
  X : ensemble de symboles de variable
  $\mathcal{F} = \mathcal{F}_0 \cup \mathcal{F}_1 \cup \mathcal{F}_2 \cup \mathcal{F}_3 \cup \cdots = \bigcup_{n \geq 0} \mathcal{F}_n$: ensemble de symboles
- **définition inductive de l'ensemble $\mathcal{T}(X, \mathcal{F})$ des termes avec variable**
  - toute variable $x \in X$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - toute constante $k \in \mathcal{F}_0$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
  - si $f \in \mathcal{F}_n$ et si $t_1, \cdots, t_n$ sont des termes dans $\mathcal{T}(X, \mathcal{F})$, alors $f(t_1, \cdots, t_n)$ est un terme dans $\mathcal{T}(X, \mathcal{F})$
- **définition inductive (récursive) de fonction**
  - terme $t[x := t']$: remplacer x par t' dans t

$$t[x := t'] = \begin{cases} y & \text{si } t = y \in X \text{ et } x \neq y \\ t' & \text{si } t = x \in X \\ k & \text{si } t = k \in \mathcal{F}_0 \\ f(t_1[x := t'], \cdots, t_n[x := t']) & \text{si } t = f(t_1, \cdots, t_n) \end{cases}$$

Langage de termes

- $\text{taille}(t) = \begin{cases} 1 & \text{si } t = x \in X \text{ ou } t = k \in \mathcal{F}_0 \\ 1 + \sum_{i=1}^{i=n} \text{taille}(t_i) & \text{si } t = f(t_1, \cdots, t_n) \end{cases}$

Langage de termes

- $\text{taille}(t) = \begin{cases} 1 & \text{si } t = x \in X \text{ ou } t = k \in \mathcal{F}_0 \\ 1 + \sum_{i=1}^{i=n} \text{taille}(t_i) & \text{si } t = f(t_1, \cdots, t_n) \end{cases}$
- raisonnement par induction structurelle : si
  - $P(x)$ est vrai pour toute variable $x \in X$
  - $P(k)$ est vrai pour toute constante $k \in \mathcal{F}_0$
  - pour tout $f \in \mathcal{F}_n$ et tous termes $t_1, \cdots, t_n$, si $P(t_1), \cdots, P(t_n)$ sont vrais, alors $P(f(t_1, \cdots, t_n))$ est vrai

  alors pour tout terme $t \in \mathcal{T}(X, \mathcal{F})$, $P(t)$ est vrai.

Langage de termes

- $\text{taille}(t) = \begin{cases} 1 & \text{si } t = x \in X \text{ ou } t = k \in \mathcal{F}_0 \\ 1 + \sum_{i=1}^{i=n} \text{taille}(t_i) & \text{si } t = f(t_1, \cdots, t_n) \end{cases}$
- raisonnement par induction structurelle : si
  - $P(x)$ est vrai pour toute variable $x \in X$
  - $P(k)$ est vrai pour toute constante $k \in \mathcal{F}_0$
  - pour tout $f \in \mathcal{F}_n$ et tous termes $t_1, \cdots, t_n$, si $P(t_1), \cdots, P(t_n)$ sont vrais, alors $P(f(t_1, \cdots, t_n))$ est vrai

  alors pour tout terme $t \in \mathcal{T}(X, \mathcal{F})$, $P(t)$ est vrai.
- pour tout terme t, $\text{taille}(t) \leq \text{taille}(t[x := t'])$

Langage de termes

- $\text{taille}(t) = \begin{cases} 1 & \text{si } t = x \in X \text{ ou } t = k \in \mathcal{F}_0 \\ 1 + \sum_{i=1}^{i=n} \text{taille}(t_i) & \text{si } t = f(t_1, \cdots, t_n) \end{cases}$
- raisonnement par induction structurelle : si
  - $P(x)$ est vrai pour toute variable $x \in X$
  - $P(k)$ est vrai pour toute constante $k \in \mathcal{F}_0$
  - pour tout $f \in \mathcal{F}_n$ et tous termes $t_1, \cdots, t_n$, si $P(t_1), \cdots, P(t_n)$ sont vrais, alors $P(f(t_1, \cdots, t_n))$ est vrai

  alors pour tout terme $t \in \mathcal{T}(X, \mathcal{F})$, $P(t)$ est vrai.
- pour tout terme t, $\text{taille}(t) \leq \text{taille}(t[x := t'])$
  - si t est un symbole de variable
    - ★ si $t = z \neq x$ alors $z[x := t'] = z$ et donc $\text{taille}(z) \leq \text{taille}(z) = \text{taille}(z[x := t'])$

Langage de termes

- $\mathrm{taille}(t) = \begin{cases} 1 & \text{si } t = x \in X \text{ ou } t = k \in \mathcal{F}_0 \\ 1 + \sum_{i=1}^{i=n} \mathrm{taille}(t_i) & \text{si } t = f(t_1, \cdots, t_n) \end{cases}$
- raisonnement par induction structurelle : si
  - $P(x)$ est vrai pour toute variable $x \in X$
  - $P(k)$ est vrai pour toute constante $k \in \mathcal{F}_0$
  - pour tout $f \in \mathcal{F}_n$ et tous termes $t_1, \cdots, t_n$, si $P(t_1), \cdots, P(t_n)$ sont vrais, alors $P(f(t_1, \cdots, t_n))$ est vrai

  alors pour tout terme $t \in \mathcal{T}(X, \mathcal{F})$, $P(t)$ est vrai.
- pour tout terme t, $\mathrm{taille}(t) \leq \mathrm{taille}(t[x := t'])$
  - si t est un symbole de variable
    - ★ si $t = z \neq x$ alors $z[x := t'] = z$ et donc $\mathrm{taille}(z) \leq \mathrm{taille}(z) = \mathrm{taille}(z[x := t'])$
    - ★ si $t = x$ alors $x[x := t'] = t'$ et donc $\mathrm{taille}(x) = 1 \leq \mathrm{taille}(t')$

Langage de termes

- $\mathrm{taille}(t) = \begin{cases} 1 & \text{si } t = x \in X \text{ ou } t = k \in \mathcal{F}_0 \\ 1 + \sum_{i=1}^{i=n} \mathrm{taille}(t_i) & \text{si } t = f(t_1, \cdots, t_n) \end{cases}$
- raisonnement par induction structurelle : si
  - $P(x)$ est vrai pour toute variable $x \in X$
  - $P(k)$ est vrai pour toute constante $k \in \mathcal{F}_0$
  - pour tout $f \in \mathcal{F}_n$ et tous termes $t_1, \cdots, t_n$, si $P(t_1), \cdots, P(t_n)$ sont vrais, alors $P(f(t_1, \cdots, t_n))$ est vrai

  alors pour tout terme $t \in \mathcal{T}(X, \mathcal{F})$, $P(t)$ est vrai.
- pour tout terme t, $\mathrm{taille}(t) \leq \mathrm{taille}(t[x := t'])$
  - si $t = k \in \mathcal{F}_0$ est un symbole de constante alors $k[x := t'] = k$ et donc $\mathrm{taille}(k) \leq \mathrm{taille}(k) = \mathrm{taille}(k[x := t'])$

Langage de termes

- $\text{taille}(t) = \begin{cases} 1 & \text{si } t = x \in X \text{ ou } t = k \in \mathcal{F}_0 \\ 1 + \sum_{i=1}^{i=n} \text{taille}(t_i) & \text{si } t = f(t_1, \cdots, t_n) \end{cases}$
- raisonnement par induction structurelle : si
  - $P(x)$ est vrai pour toute variable $x \in X$
  - $P(k)$ est vrai pour toute constante $k \in \mathcal{F}_0$
  - pour tout $f \in \mathcal{F}_n$ et tous termes $t_1, \cdots, t_n$, si $P(t_1), \cdots, P(t_n)$ sont vrais, alors $P(f(t_1, \cdots, t_n))$ est vrai

  alors pour tout terme $t \in \mathcal{T}(X, \mathcal{F})$, $P(t)$ est vrai.
- pour tout terme t, $\text{taille}(t) \leq \text{taille}(t[x := t'])$
  - si $t = f(t_1, \cdots, t_n)$ alors $f(t_1, \cdots, t_n)[x := t'] = f(t_1[x := t'], \cdots, t_n[x := t'])$ et donc

$$\begin{array}{ll} & \text{taille}(f(t_1, \cdots, t_n)) = 1 + \sum_{i=1}^{i=n} \text{taille}(t_i) \\ \leq & 1 + \sum_{i=1}^{i=n} \text{taille}(t_i[x := t']) \\ & \text{hyp. induction : } \text{taille}(t_i) \leq \text{taille}(t_i[x := t']) \\ & = \text{taille}(f(t_1, \cdots, t_n)[x := t']) \end{array}$$

Langage de termes

- $\text{taille}(t) = \begin{cases} 1 & \text{si } t = x \in X \text{ ou } t = k \in \mathcal{F}_0 \\ 1 + \sum_{i=1}^{i=n} \text{taille}(t_i) & \text{si } t = f(t_1, \cdots, t_n) \end{cases}$
- raisonnement par induction structurelle : si
  - $P(x)$ est vrai pour toute variable $x \in X$
  - $P(k)$ est vrai pour toute constante $k \in \mathcal{F}_0$
  - pour tout $f \in \mathcal{F}_n$ et tous termes $t_1, \cdots, t_n$, si $P(t_1), \cdots, P(t_n)$ sont vrais, alors $P(f(t_1, \cdots, t_n))$ est vrai

  alors pour tout terme $t \in \mathcal{T}(X, \mathcal{F})$, $P(t)$ est vrai.
- pour tout terme t, $\text{taille}(t) \leq \text{taille}(t[x := t'])$

Langage des formules atomiques

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **termes** sur lesquels il s'applique
  - $\mathcal{P} = \mathcal{P}_0 \cup \mathcal{P}_1 \cup \mathcal{P}_2 \cup \cdots = \bigcup_{n \geq 0} \mathcal{P}_n$: ensemble des symboles de prédicat

Langage des formules atomiques

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **termes** sur lesquels il s'applique
  - $\mathcal{P} = \mathcal{P}_0 \cup \mathcal{P}_1 \cup \mathcal{P}_2 \cup \cdots = \bigcup_{n \geq 0} \mathcal{P}_n$: ensemble des symboles de prédicat

    $\mathcal{P}_0$ contient les symboles de proposition

Langage des formules atomiques

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **termes** sur lesquels il s'applique
  - $\mathcal{P} = \mathcal{P}_0 \cup \mathcal{P}_1 \cup \mathcal{P}_2 \cup \cdots = \bigcup_{n \geq 0} \mathcal{P}_n$: ensemble des symboles de prédicat

    $\mathcal{P}_0$ contient les symboles de proposition
    $\mathcal{P}_n$ contient les symboles de prédicat d'arité n

Langage des formules atomiques

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **termes** sur lesquels il s'applique
  - $\mathcal{P} = \mathcal{P}_0 \cup \mathcal{P}_1 \cup \mathcal{P}_2 \cup \cdots = \bigcup_{n \geq 0} \mathcal{P}_n$: ensemble des symboles de prédicat

    $\mathcal{P}_0$ contient les symboles de proposition
    $\mathcal{P}_n$ contient les symboles de prédicat d'arité n
- ensemble $\mathcal{L}(X, \mathcal{F}, \mathcal{P})$ des **formules atomiques avec variables**

Langage des formules atomiques

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **termes** sur lesquels il s'applique
  - $\mathcal{P} = \mathcal{P}_0 \cup \mathcal{P}_1 \cup \mathcal{P}_2 \cup \cdots = \bigcup_{n \geq 0} \mathcal{P}_n$: ensemble des symboles de prédicat

    $\mathcal{P}_0$ contient les symboles de proposition
    $\mathcal{P}_n$ contient les symboles de prédicat d'arité n

- ensemble $\mathcal{L}(X, \mathcal{F}, \mathcal{P})$ des **formules atomiques avec variables**
  - les symboles de $\mathcal{P}_0$ sont des formules atomiques de $\mathcal{L}(X, \mathcal{F}, \mathcal{P})$

Langage des formules atomiques

- les énoncés atomiques sont décomposés en un symbole de **prédicat** et les **termes** sur lesquels il s'applique
  - $\mathcal{P} = \mathcal{P}_0 \cup \mathcal{P}_1 \cup \mathcal{P}_2 \cup \cdots = \bigcup_{n \geq 0} \mathcal{P}_n$: ensemble des symboles de prédicat

    $\mathcal{P}_0$ contient les symboles de proposition
    $\mathcal{P}_n$ contient les symboles de prédicat d'arité n

- ensemble $\mathcal{L}(X, \mathcal{F}, \mathcal{P})$ des **formules atomiques avec variables**
  - les symboles de $\mathcal{P}_0$ sont des formules atomiques de $\mathcal{L}(X, \mathcal{F}, \mathcal{P})$
  - si p est un symbole de prédicat d'arité n ($p \in \mathcal{P}_n$) et si $t_1, \cdots, t_n$ sont des termes dans $\mathcal{T}(X, \mathcal{F})$ alors $p(t_1, \cdots, t_n)$ est une formule atomique de $\mathcal{L}(X, \mathcal{F}, \mathcal{P})$

Langage des formules logiques

- **définition inductive de l'ensemble $\mathbb{F}(X, \mathcal{F}, \mathcal{P})$ des formules logiques avec variables**

Langage des formules logiques

- **définition inductive de l'ensemble $\mathbb{F}(X, \mathcal{F}, \mathcal{P})$ des formules logiques avec variables**
  - toute formule atomique de $\mathcal{L}(X, \mathcal{F}, \mathcal{P})$ est une formule logique

Langage des formules logiques

- **définition inductive de l'ensemble $\mathbb{F}(X, \mathcal{F}, \mathcal{P})$ des formules logiques avec variables**
  - toute formule atomique de $\mathcal{L}(X, \mathcal{F}, \mathcal{P})$ est une formule logique
  - true et false sont des formules logiques

Langage des formules logiques

- **définition inductive de l'ensemble $\mathbb{F}(X, \mathcal{F}, \mathcal{P})$ des formules logiques avec variables**
  - toute formule atomique de $\mathcal{L}(X, \mathcal{F}, \mathcal{P})$ est une formule logique
  - true et false sont des formules logiques
  - si F est une formule logique, alors $\neg F$ est aussi une formule logique

Langage des formules logiques

- **définition inductive de l'ensemble $\mathbb{F}(X, \mathcal{F}, \mathcal{P})$ des formules logiques avec variables**
  - toute formule atomique de $\mathcal{L}(X, \mathcal{F}, \mathcal{P})$ est une formule logique
  - true et false sont des formules logiques
  - si F est une formule logique, alors $\neg F$ est aussi une formule logique
  - si F_1 et F_2 sont des formules logiques, alors $(F_1 \wedge F_2)$, $(F_1 \vee F_2)$ et $(F_1 \Rightarrow F_2)$ sont aussi des formules logiques

Langage des formules logiques

- **définition inductive de l'ensemble $\mathbb{F}(X, \mathcal{F}, \mathcal{P})$ des formules logiques avec variables**
  - toute formule atomique de $\mathcal{L}(X, \mathcal{F}, \mathcal{P})$ est une formule logique
  - true et false sont des formules logiques
  - si F est une formule logique, alors $\neg F$ est aussi une formule logique
  - si F_1 et F_2 sont des formules logiques, alors $(F_1 \wedge F_2)$, $(F_1 \vee F_2)$ et $(F_1 \Rightarrow F_2)$ sont aussi des formules logiques
  - si F est une formule logique et x un symbole de variable, alors $\forall x\, F$ et $\exists x\, F$ sont aussi des formules logiques

Langage des formules logiques

- **définition inductive de l'ensemble $\mathbb{F}(X, \mathcal{F}, \mathcal{P})$ des formules logiques avec variables**
  - toute formule atomique de $\mathcal{L}(X, \mathcal{F}, \mathcal{P})$ est une formule logique
  - true et false sont des formules logiques
  - si F est une formule logique, alors $\neg F$ est aussi une formule logique
  - si F_1 et F_2 sont des formules logiques, alors $(F_1 \wedge F_2)$, $(F_1 \vee F_2)$ et $(F_1 \Rightarrow F_2)$ sont aussi des formules logiques
  - si F est une formule logique et x un symbole de variable, alors $\forall x\, F$ et $\exists x\, F$ sont aussi des formules logiques

  toutes les formules logiques sont obtenues en appliquant un nombre fini de fois ces règles de construction

Variables libres – Variables liées

- $p(x)$ exprime que l'objet désigné par x vérifie la propriété p

Variables libres – Variables liées

- $p(x)$ exprime que l'objet désigné par x vérifie la propriété p
  - pour déterminer si cette formule est « vraie », il faut connaître l'objet désigné par x

Variables libres – Variables liées

- $p(x)$ exprime que l'objet désigné par x vérifie la propriété p
  - pour déterminer si cette formule est « vraie », il faut connaître l'objet désigné par x
    - information extérieure fournie par une valuation (un environnement)

Variables libres – Variables liées

- $p(x)$ exprime que l'objet désigné par x vérifie la propriété p
  - pour déterminer si cette formule est « vraie », il faut connaître l'objet désigné par x
    - information extérieure fournie par une valuation (un environnement)
  - x : **occurrence de variable libre**

Variables libres – Variables liées

- $p(x)$ exprime que l'objet désigné par x vérifie la propriété p
  - pour déterminer si cette formule est « vraie », il faut connaître l'objet désigné par x
    - information extérieure fournie par une valuation (un environnement)
  - x : **occurrence de variable libre**
  - les formules $p(x)$ et $p(y)$ correspondent à des énoncés différents
    - les objets désignés par x et par y ne vérifient pas nécessairement les mêmes propriétés

Variables libres – Variables liées

- $p(x)$ exprime que l'objet désigné par x vérifie la propriété p
  - pour déterminer si cette formule est « vraie », il faut connaître l'objet désigné par x
    - information extérieure fournie par une valuation (un environnement)
  - x : **occurrence de variable libre**
  - les formules $p(x)$ et $p(y)$ correspondent à des énoncés différents
    - les objets désignés par x et par y ne vérifient pas nécessairement les mêmes propriétés
- $\forall x\ p(x)$ exprime que la propriété p est vérifiée par tous les objets de l'univers du discours

Variables libres – Variables liées

- $p(x)$ exprime que l'objet désigné par x vérifie la propriété p
  - pour déterminer si cette formule est « vraie », il faut connaître l'objet désigné par x
    - information extérieure fournie par une valuation (un environnement)
  - x : **occurrence de variable libre**
  - les formules $p(x)$ et $p(y)$ correspondent à des énoncés différents
    - les objets désignés par x et par y ne vérifient pas nécessairement les mêmes propriétés
- $\forall x\ p(x)$ exprime que la propriété p est vérifiée par tous les objets de l'univers du discours
  - pour déterminer si cette formule est « vraie », il faut vérifier la propriété p pour tous les objets

Variables libres – Variables liées

- $p(x)$ exprime que l'objet désigné par x vérifie la propriété p
  - pour déterminer si cette formule est « vraie », il faut connaître l'objet désigné par x
    - information extérieure fournie par une valuation (un environnement)
  - x : **occurrence de variable libre**
  - les formules $p(x)$ et $p(y)$ correspondent à des énoncés différents
    - les objets désignés par x et par y ne vérifient pas nécessairement les mêmes propriétés
- $\forall x\ p(x)$ exprime que la propriété p est vérifiée par tous les objets de l'univers du discours
  - pour déterminer si cette formule est « vraie », il faut vérifier la propriété p pour tous les objets
  - x : **occurrence de variable liée**

Variables libres – Variables liées

- $p(x)$ exprime que l'objet désigné par x vérifie la propriété p
  - pour déterminer si cette formule est « vraie », il faut connaître l'objet désigné par x
    - information extérieure fournie par une valuation (un environnement)
  - x : **occurrence de variable libre**
  - les formules $p(x)$ et $p(y)$ correspondent à des énoncés différents
    - les objets désignés par x et par y ne vérifient pas nécessairement les mêmes propriétés
- $\forall x \; p(x)$ exprime que la propriété p est vérifiée par tous les objets de l'univers du discours
  - pour déterminer si cette formule est « vraie », il faut vérifier la propriété p pour tous les objets
  - x : **occurrence de variable liée**
  - les formules $\forall x \; p(x)$ et $\forall y \; p(y)$ ont la même signification
    - l'occurrence liée de x est une occurrence de **variable muette**, on peut la renommer

Variables libres – Variables liées

- $p(x)$ exprime que l'objet désigné par x vérifie la propriété p
  - pour déterminer si cette formule est « vraie », il faut connaître l'objet désigné par x
    - information extérieure fournie par une valuation (un environnement)
  - x : **occurrence de variable libre**
  - les formules $p(x)$ et $p(y)$ correspondent à des énoncés différents
    - les objets désignés par x et par y ne vérifient pas nécessairement les mêmes propriétés
- $\exists x\ p(x)$ exprime que la propriété p est vérifiée par au moins un objet de l'univers du discours

Variables libres – Variables liées

- $p(x)$ exprime que l'objet désigné par x vérifie la propriété p
  - pour déterminer si cette formule est « vraie », il faut connaître l'objet désigné par x
    - information extérieure fournie par une valuation (un environnement)
  - x : **occurrence de variable libre**
  - les formules $p(x)$ et $p(y)$ correspondent à des énoncés différents
    - les objets désignés par x et par y ne vérifient pas nécessairement les mêmes propriétés
- $\exists x\ p(x)$ exprime que la propriété p est vérifiée par au moins un objet de l'univers du discours
  - pour déterminer si cette formule est « vraie », il faut parcourir l'ensemble des objets pour vérifier s'il en existe un qui vérifie la propriété p

Variables libres – Variables liées

- $p(x)$ exprime que l'objet désigné par x vérifie la propriété p
  - pour déterminer si cette formule est « vraie », il faut connaître l'objet désigné par x
    - information extérieure fournie par une valuation (un environnement)
  - x : **occurrence de variable libre**
  - les formules $p(x)$ et $p(y)$ correspondent à des énoncés différents
    - les objets désignés par x et par y ne vérifient pas nécessairement les mêmes propriétés
- $\exists x \; p(x)$ exprime que la propriété p est vérifiée par au moins un objet de l'univers du discours
  - pour déterminer si cette formule est « vraie », il faut parcourir l'ensemble des objets pour vérifier s'il en existe un qui vérifie la propriété p
  - x : **occurrence de variable liée**

Variables libres – Variables liées

- $p(x)$ exprime que l'objet désigné par x vérifie la propriété p
  - pour déterminer si cette formule est « vraie », il faut connaître l'objet désigné par x
    - information extérieure fournie par une valuation (un environnement)
  - x : **occurrence de variable libre**
  - les formules $p(x)$ et $p(y)$ correspondent à des énoncés différents
    - les objets désignés par x et par y ne vérifient pas nécessairement les mêmes propriétés
- $\exists x\ p(x)$ exprime que la propriété p est vérifiée par au moins un objet de l'univers du discours
  - pour déterminer si cette formule est « vraie », il faut parcourir l'ensemble des objets pour vérifier s'il en existe un qui vérifie la propriété p
  - x : **occurrence de variable liée**
  - les formules $\exists x\ p(x)$ et $\exists y\ p(y)$ ont la même signification
    - x : occurrence de **variable muette**, on peut la renommer

Variables libres – Variables liées

- une occurrence de variable est **libre** si elle n'est pas dans la portée d'un quantificateur (sinon elle est **liée**)

Variables libres – Variables liées

- une occurrence de variable est **libre** si elle n'est pas dans la portée d'un quantificateur (sinon elle est **liée**)
  - dans la formule $p(x) \wedge \forall x\ q(x)$ la première occurrence de x est libre tandis que la deuxième occurrence de x est liée

Variables libres – Variables liées

- une occurrence de variable est **libre** si elle n'est pas dans la portée d'un quantificateur (sinon elle est **liée**)
- les **variables libres d'une formule** sont les variables qui ont au moins une occurrence libre dans la formule

Variables libres – Variables liées

- une occurrence de variable est **libre** si elle n'est pas dans la portée d'un quantificateur (sinon elle est **liée**)
- les **variables libres d'une formule** sont les variables qui ont au moins une occurrence libre dans la formule

Variables libres – Variables liées

- une occurrence de variable est **libre** si elle n'est pas dans la portée d'un quantificateur (sinon elle est **liée**)
- les **variables libres d'une formule** sont les variables qui ont au moins une occurrence libre dans la formule
- variables libres d'une formule $\mathrm{Free} : \mathbb{F}(X, \mathcal{F}, \mathcal{P}) \to \wp(X)$

$$\mathrm{Free}(F) = \left\{ \begin{array}{l} \\ \\ \\ \end{array} \right.$$

Variables libres – Variables liées

- une occurrence de variable est **libre** si elle n'est pas dans la portée d'un quantificateur (sinon elle est **liée**)
- les **variables libres d'une formule** sont les variables qui ont au moins une occurrence libre dans la formule
- variables libres d'une formule $\mathrm{Free} : \mathbb{F}(X, \mathcal{F}, \mathcal{P}) \to \wp(X)$

$$\mathrm{Free}(F) = \begin{cases} \emptyset & \text{si } F = \text{true ou } F = \text{false} \\ & \\ & \\ & \end{cases}$$

Variables libres – Variables liées

- une occurrence de variable est **libre** si elle n'est pas dans la portée d'un quantificateur (sinon elle est **liée**)
- les **variables libres d'une formule** sont les variables qui ont au moins une occurrence libre dans la formule
- variables apparaissant dans un terme $\vartheta : \mathcal{T}(X, \mathcal{F}) \to \wp(X)$
- variables libres d'une formule $\mathrm{Free} : \mathbb{F}(X, \mathcal{F}, \mathcal{P}) \to \wp(X)$

$$\mathrm{Free}(F) = \begin{cases} \emptyset & \text{si } F = \text{true ou } F = \text{false} \\ \bigcup_{i=1}^{n} \vartheta(t_i) & \text{si } F = p(t_1, \cdots, t_n) \end{cases}$$

Variables libres – Variables liées

- une occurrence de variable est **libre** si elle n'est pas dans la portée d'un quantificateur (sinon elle est **liée**)
- les **variables libres d'une formule** sont les variables qui ont au moins une occurrence libre dans la formule
- variables apparaissant dans un terme $\vartheta : \mathcal{T}(X, \mathcal{F}) \rightarrow \wp(X)$
- variables libres d'une formule $\mathrm{Free} : \mathbb{F}(X, \mathcal{F}, \mathcal{P}) \rightarrow \wp(X)$

$$\mathrm{Free}(F) = \begin{cases} \emptyset & \text{si } F = \text{true ou } F = \text{false} \\ \bigcup_{i=1}^{n} \vartheta(t_i) & \text{si } F = p(t_1, \cdots, t_n) \\ \mathrm{Free}(F') & \text{si } F = \neg F' \\ & \\ & \end{cases}$$

Variables libres – Variables liées

- une occurrence de variable est **libre** si elle n'est pas dans la portée d'un quantificateur (sinon elle est **liée**)
- les **variables libres d'une formule** sont les variables qui ont au moins une occurrence libre dans la formule
- variables apparaissant dans un terme $\vartheta : \mathcal{T}(X, \mathcal{F}) \rightarrow \wp(X)$
- variables libres d'une formule $\mathrm{Free} : \mathbb{F}(X, \mathcal{F}, \mathcal{P}) \rightarrow \wp(X)$

$$
\mathrm{Free}(F) = \begin{cases}
\emptyset & \text{si } F = \text{true ou } F = \text{false} \\
\bigcup_{i=1}^{n} \vartheta(t_i) & \text{si } F = p(t_1, \cdots, t_n) \\
\mathrm{Free}(F') & \text{si } F = \neg F' \\
\mathrm{Free}(F_1) \cup \mathrm{Free}(F_2) & \text{si } F = F_1 \wedge F_2 \text{ ou } F = F_1 \vee F_2 \\
 & \quad \text{ou } F = F_1 \Rightarrow F_2
\end{cases}
$$

Variables libres – Variables liées

- une occurrence de variable est **libre** si elle n'est pas dans la portée d'un quantificateur (sinon elle est **liée**)
- les **variables libres d'une formule** sont les variables qui ont au moins une occurrence libre dans la formule
- variables apparaissant dans un terme $\vartheta : \mathcal{T}(X, \mathcal{F}) \to \wp(X)$
- variables libres d'une formule $\mathrm{Free} : \mathbb{F}(X, \mathcal{F}, \mathcal{P}) \to \wp(X)$

$$\mathrm{Free}(F) = \begin{cases} \emptyset & \text{si } F = \text{true ou } F = \text{false} \\ \bigcup_{i=1}^{n} \vartheta(t_i) & \text{si } F = p(t_1, \cdots, t_n) \\ \mathrm{Free}(F') & \text{si } F = \neg F' \\ \mathrm{Free}(F_1) \cup \mathrm{Free}(F_2) & \text{si } F = F_1 \wedge F_2 \text{ ou } F = F_1 \vee F_2 \\ & \quad \text{ou } F = F_1 \Rightarrow F_2 \\ \mathrm{Free}(F') \backslash \{x\} & \text{si } F = \forall x\ F' \text{ ou } F = \exists x\ F' \end{cases}$$

Variables libres – Variables liées

- une occurrence de variable est **libre** si elle n'est pas dans la portée d'un quantificateur (sinon elle est **liée**)
- les **variables libres d'une formule** sont les variables qui ont au moins une occurrence libre dans la formule
- *exemple :* $\forall x\,(p(x,y) \Rightarrow \exists y\,(q(y) \wedge r(x,y,z)))$

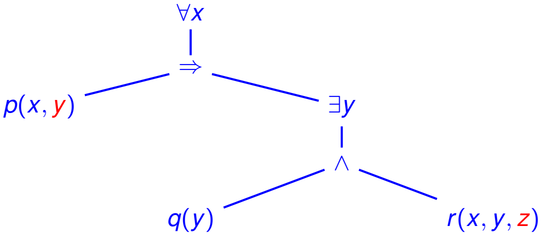
Variables libres – Variables liées

- une occurrence de variable est **libre** si elle n'est pas dans la portée d'un quantificateur (sinon elle est **liée**)
- les **variables libres d'une formule** sont les variables qui ont au moins une occurrence libre dans la formule
- *exemple :* $\forall x\,(p(x,y) \Rightarrow \exists y\,(q(y) \wedge r(x,y,z)))$

$$
\begin{array}{rl}
 & \mathrm{Free}(\forall x\,(p(x,y) \Rightarrow \exists y\,(q(y) \wedge r(x,y,z)))) \\
= & \mathrm{Free}(p(x,y) \Rightarrow \exists y\,(q(y) \wedge r(x,y,z))) \setminus \{x\} \\
= & \mathrm{Free}(p(x,y) \Rightarrow \exists y\,(q(y) \wedge r(x,y,z))) \setminus \{x\} \\
= & (\mathrm{Free}(p(x,y)) \cup \mathrm{Free}(\exists y\,(q(y) \wedge r(x,y,z)))) \setminus \{x\} \\
= & (\{x,y\} \cup \mathrm{Free}(\exists y\,(q(y) \wedge r(x,y,z)))) \setminus \{x\} \\
= & (\{x,y\} \cup (\mathrm{Free}((q(y) \wedge r(x,y,z))) \setminus \{y\})) \setminus \{x\} \\
= & (\{x,y\} \cup (\mathrm{Free}(q(y)) \cup \mathrm{Free}(r(x,y,z)) \setminus \{y\})) \setminus \{x\} \\
= & (\{x,y\} \cup (\{y\} \cup \{x,y,z\} \setminus \{y\})) \setminus \{x\} \\
= & (\{x,y\} \cup \{x,z\}) \setminus \{x\} \\
= & \{y,z\}
\end{array}
$$

Variables libres – Variables liées

- une occurrence de variable est **libre** si elle n'est pas dans la portée d'un quantificateur (sinon elle est **liée**)
- les **variables libres d'une formule** sont les variables qui ont au moins une occurrence libre dans la formule
- *exemple :* $\forall x\,(p(x,y) \Rightarrow \exists y\,(q(y) \wedge r(x,y,z)))$



Variables libres – Variables liées

- une occurrence de variable est **libre** si elle n'est pas dans la portée d'un quantificateur (sinon elle est **liée**)
- les **variables libres d'une formule** sont les variables qui ont au moins une occurrence libre dans la formule
- *exemple :* $\forall x\,(p(x, y) \Rightarrow \exists y\,(q(y) \wedge r(x, y, z)))$
  - la variable y admet une occurrence libre et deux occurrences liées

Variables libres – Variables liées

- une occurrence de variable est **libre** si elle n'est pas dans la portée d'un quantificateur (sinon elle est **liée**)
- les **variables libres d'une formule** sont les variables qui ont au moins une occurrence libre dans la formule
- *exemple :* $\forall x\,(p(x,y) \Rightarrow \exists y\,(q(y) \wedge r(x,y,z)))$
  - la variable y admet une occurrence libre et deux occurrences liées
  - pour éviter qu'un même symbole de variable admette des occurrences libres et liées au sein d'une formule on renomme les occurrences liées avec un symbole qui n'a pas d'occurrence libre

$$\forall x\,(p(x,y) \Rightarrow \exists w\,(q(w) \wedge r(x,w,z)))$$

Variables libres – Variables liées

- une occurrence de variable est **libre** si elle n'est pas dans la portée d'un quantificateur (sinon elle est **liée**)
- les **variables libres d'une formule** sont les variables qui ont au moins une occurrence libre dans la formule
- *exemple :* $\forall x\,(p(x,y) \Rightarrow \exists y\,(q(y) \wedge r(x,y,z)))$
  - la variable y admet une occurrence libre et deux occurrences liées
  - pour éviter qu'un même symbole de variable admette des occurrences libres et liées au sein d'une formule on renomme les occurrences liées avec un symbole qui n'a pas d'occurrence libre

$$\forall x\,(p(x,y) \Rightarrow \exists w\,(q(w) \wedge r(x,w,z)))$$

  - l'ensemble des variables libres de cette nouvelle formule ne change pas (c'est toujours $\{y,z\}$)

Formules closes

- une **formule close** est une formule F telle que $\mathrm{Free}(F) = \emptyset$

Formules closes

- une **formule close** est une formule F telle que $\mathrm{Free}(F) = \emptyset$
- si $\mathrm{Free}(F) = \{x_1, \cdots, x_n\}$, **une clôture universelle** de F est la formule $\forall x_1 \cdots \forall x_n\, F$

Formules closes

- une **formule close** est une formule F telle que $\mathrm{Free}(F) = \emptyset$
- si $\mathrm{Free}(F) = \{x_1, \cdots, x_n\}$, **une clôture universelle** de F est la formule $\forall x_1 \cdots \forall x_n\, F$
  - *exemple :* $F = \forall x\,(p(x, y) \Rightarrow \exists w\,(q(w) \wedge r(x, w, z)))$
    $\mathrm{Free}(F) = \{y, z\}$

Formules closes

- une **formule close** est une formule F telle que $\mathrm{Free}(F) = \emptyset$
- si $\mathrm{Free}(F) = \{x_1, \cdots, x_n\}$, **une clôture universelle** de F est la formule $\forall x_1 \cdots \forall x_n \, F$
  - *exemple :* $F = \forall x \, (p(x, y) \Rightarrow \exists w \, (q(w) \wedge r(x, w, z)))$
    $\mathrm{Free}(F) = \{y, z\}$
    clôture universelle de F :

$$\forall y \, \forall z \, \forall x \, (p(x, y) \Rightarrow \exists w \, (q(w) \wedge r(x, w, z)))$$

Formules closes

- une **formule close** est une formule F telle que $\mathrm{Free}(F) = \emptyset$
- si $\mathrm{Free}(F) = \{x_1, \cdots, x_n\}$, **une clôture universelle** de F est la formule $\forall x_1 \cdots \forall x_n\, F$
  - *exemple :* $F = \forall x\,(p(x,y) \Rightarrow \exists w\,(q(w) \wedge r(x,w,z)))$
    $\mathrm{Free}(F) = \{y, z\}$
    clôture universelle de F :

    $$\forall y\, \forall z\, \forall x\,(p(x,y) \Rightarrow \exists w\,(q(w) \wedge r(x,w,z)))$$

    clôture universelle de F :

    $$\forall z\, \forall y\, \forall x\,(p(x,y) \Rightarrow \exists w\,(q(w) \wedge r(x,w,z)))$$

Substitution dans une formule

- substituer une variable x par un terme t dans une formule F pour obtenir une formule notée $F[x := t]$ consiste à remplacer certaines occurrences de x dans F par le terme t : toutes les occurrences ?

Substitution dans une formule

- substituer une variable x par un terme t dans une formule F pour obtenir une formule notée $F[x := t]$ consiste à remplacer certaines occurrences de x dans F par le terme t : toutes les occurrences ?
- à partir de deux formules F_1 et F_2 qui expriment la même propriété :

$$F_1 : \quad (\forall x\ p(x)) \vee (\exists z\ q(x, z))$$
$$F_2 : \quad (\forall w\ p(w)) \vee (\exists z\ q(x, z))$$

Substitution dans une formule

- substituer une variable x par un terme t dans une formule F pour obtenir une formule notée $F[x := t]$ consiste à remplacer certaines occurrences de x dans F par le terme t : toutes les occurrences ?
- à partir de deux formules F_1 et F_2 qui expriment la même propriété :

  $$F_1 : \quad (\forall x\ p(x)) \vee (\exists z\ q(x, z))$$
  $$F_2 : \quad (\forall w\ p(w)) \vee (\exists z\ q(x, z))$$

  en substituant x par $f(v)$ on souhaite obtenir des formules logiquement équivalentes

Substitution dans une formule

- substituer une variable x par un terme t dans une formule F pour obtenir une formule notée $F[x := t]$ consiste à remplacer certaines occurrences de x dans F par le terme t : toutes les occurrences ?
- à partir de deux formules F_1 et F_2 qui expriment la même propriété :

  $$F_1 : \quad (\forall x\ p(x)) \vee (\exists z\ q(x, z))$$
  $$F_2 : \quad (\forall w\ p(w)) \vee (\exists z\ q(x, z))$$

  en substituant x par $f(v)$ on souhaite obtenir des formules logiquement équivalentes

  - ce n'est pas le cas si l'on substitue toutes les occurrences

    $$F_1 : \quad (\forall x\ p(\boxed{f(v)})) \vee (\exists z\ q(f(v), z))$$
    $$F_2 : \quad (\forall w\ p(\boxed{w})) \vee (\exists z\ q(f(v), z))$$

Substitution dans une formule

- substituer une variable x par un terme t dans une formule F pour obtenir une formule notée $F[x := t]$ consiste à remplacer certaines occurrences de x dans F par le terme t : toutes les occurrences ?
- à partir de deux formules F_1 et F_2 qui expriment la même propriété :

  $F_1 : \quad (\forall x\ p(x)) \lor (\exists z\ q(x, z))$
  $F_2 : \quad (\forall w\ p(w)) \lor (\exists z\ q(x, z))$

  en substituant x par $f(v)$ on souhaite obtenir des formules logiquement équivalentes
  - ce n'est pas le cas si l'on substitue toutes les occurrences

    $F_1 : \quad (\forall x\ p(\boxed{f(v)})) \lor (\exists z\ q(f(v), z))$
    $F_2 : \quad (\forall w\ p(\boxed{w})) \lor (\exists z\ q(f(v), z))$
- toutes les occurrences ? non ... **uniquement les occurrences libres**

$$F_1[x := f(v)] = (\forall x\ p(\underline{x})) \lor (\exists z\ q(f(v), z))$$

Substitution dans une formule

- substituer une variable x par un terme t dans une formule F pour obtenir une formule notée $F[x := t]$ consiste à remplacer les occurrences libres de x dans F par le terme t
  - correct uniquement lorsque les variables apparaissant dans t ne sont pas liées dans F : phénomène de **capture de variable**

Substitution dans une formule

- substituer une variable x par un terme t dans une formule F pour obtenir une formule notée $F[x := t]$ consiste à remplacer les occurrences libres de x dans F par le terme t
  - correct uniquement lorsque les variables apparaissant dans t ne sont pas liées dans F : phénomène de **capture de variable**
    - *exemple* : substituer y par le terme $f(x)$ dans la formule $\forall x \ p(x, y)$ en remplaçant y par $f(x)$ on obtient $\forall x \ p(x, f(x))$ ce qui est incorrect puisque l'occurrence de x dans le terme $f(x)$ devient liée

Substitution dans une formule

- substituer une variable x par un terme t dans une formule F pour obtenir une formule notée $F[x := t]$ consiste à remplacer les occurrences libres de x dans F par le terme t
  - correct uniquement lorsque les variables apparaissant dans t ne sont pas liées dans F : phénomène de **capture de variable**
    - *exemple* : substituer y par le terme $f(x)$ dans la formule $\forall x\ p(x, y)$ en remplaçant y par $f(x)$ on obtient $\forall x\ p(x, f(x))$ ce qui est incorrect puisque l'occurrence de x dans le terme $f(x)$ devient liée
  - avant d'effectuer la substitution, il faut **renommer** (par de nouvelles variables) les variables liées de F qui apparaissent dans t

Substitution dans une formule

- substituer une variable x par un terme t dans une formule F pour obtenir une formule notée $F[x := t]$ consiste à remplacer les occurrences libres de x dans F par le terme t
  - correct uniquement lorsque les variables apparaissant dans t ne sont pas liées dans F : phénomène de **capture de variable**
    - *exemple* : substituer y par le terme $f(x)$ dans la formule $\forall x\ p(x, y)$ en remplaçant y par $f(x)$ on obtient $\forall x\ p(x, f(x))$ ce qui est incorrect puisque l'occurrence de x dans le terme $f(x)$ devient liée
  - avant d'effectuer la substitution, il faut **renommer** (par de nouvelles variables) les variables liées de F qui apparaissent dans t
    - *exemple* : substituer y par le terme $f(x)$ dans la formule $\forall x\ p(x, y)$

Substitution dans une formule

- substituer une variable x par un terme t dans une formule F pour obtenir une formule notée $F[x := t]$ consiste à remplacer les occurrences libres de x dans F par le terme t
  - correct uniquement lorsque les variables apparaissant dans t ne sont pas liées dans F : phénomène de **capture de variable**
    - ★ *exemple* : substituer y par le terme $f(x)$ dans la formule $\forall x\ p(x, y)$ en remplaçant y par $f(x)$ on obtient $\forall x\ p(x, f(x))$ ce qui est incorrect puisque l'occurrence de x dans le terme $f(x)$ devient liée
  - avant d'effectuer la substitution, il faut **renommer** (par de nouvelles variables) les variables liées de F qui apparaissent dans t
    - ★ *exemple* : substituer y par le terme $f(x)$ dans la formule $\forall x\ p(x, y)$

      (1) renommage de l'occurrence liée de x par z pour obtenir $\forall z\ p(z, y)$ (qui est logiquement équivalente à $\forall x\ p(x, y)$)

Substitution dans une formule

- substituer une variable x par un terme t dans une formule F pour obtenir une formule notée $F[x := t]$ consiste à remplacer les occurrences libres de x dans F par le terme t
  - correct uniquement lorsque les variables apparaissant dans t ne sont pas liées dans F : phénomène de **capture de variable**
    - *exemple* : substituer y par le terme $f(x)$ dans la formule $\forall x\ p(x, y)$ en remplaçant y par $f(x)$ on obtient $\forall x\ p(x, f(x))$ ce qui est incorrect puisque l'occurrence de x dans le terme $f(x)$ devient liée
  - avant d'effectuer la substitution, il faut **renommer** (par de nouvelles variables) les variables liées de F qui apparaissent dans t
    - *exemple* : substituer y par le terme $f(x)$ dans la formule $\forall x\ p(x, y)$

    (1) renommage de l'occurrence liée de x par z pour obtenir $\forall z\ p(z, y)$ (qui est logiquement équivalente à $\forall x\ p(x, y)$)

    (2) application de la substitution de y par $f(x)$ pour obtenir $\forall z\ p(z, f(x))$

Substitution dans une formule

définition inductive de la formule $F[x := t]$ lorsqu'aucune variable liée de F n'apparaît dans t

Substitution dans une formule

définition inductive de la formule $F[x := t]$ lorsqu'aucune variable liée de F n'apparaît dans t

$$\begin{array}{rcl} \mathrm{true}[x := t] & = & \mathrm{true} \\ \mathrm{false}[x := t] & = & \mathrm{false} \end{array}$$

Substitution dans une formule

définition inductive de la formule $F[x := t]$ lorsqu'aucune variable liée de F n'apparaît dans t

$$\begin{array}{rcl} \text{true}[x := t] & = & \text{true} \\ \text{false}[x := t] & = & \text{false} \\ p(t_1, \cdots, t_n)[x := t] & = & p(t_1[x := t], \cdots, t_n[x := t]) \quad (p \in \mathcal{P}_n) \end{array}$$

Substitution dans une formule

définition inductive de la formule $F[x := t]$ lorsqu'aucune variable liée de F n'apparaît dans t

$$
\begin{array}{rcl}
\mathrm{true}[x := t] & = & \mathrm{true} \\
\mathrm{false}[x := t] & = & \mathrm{false} \\
p(t_1, \cdots, t_n)[x := t] & = & p(t_1[x := t], \cdots, t_n[x := t]) \quad (p \in \mathcal{P}_n) \\
(\neg F_0)[x := t] & = & \neg(F_0[x := t]) \\
(F_1 \wedge F_2)[x := t] & = & F_1[x := t] \wedge F_2[x := t] \\
(F_1 \vee F_2)[x := t] & = & F_1[x := t] \vee F_2[x := t] \\
(F_1 \Rightarrow F_2)[x := t] & = & F_1[x := t] \Rightarrow F_2[x := t]
\end{array}
$$

Substitution dans une formule

définition inductive de la formule $F[x := t]$ lorsqu'aucune variable liée de F n'apparaît dans t

$$
\begin{array}{rcl}
\mathrm{true}[x := t] & = & \mathrm{true} \\
\mathrm{false}[x := t] & = & \mathrm{false} \\
p(t_1, \cdots, t_n)[x := t] & = & p(t_1[x := t], \cdots, t_n[x := t]) \quad (p \in \mathcal{P}_n) \\
(\neg F_0)[x := t] & = & \neg(F_0[x := t]) \\
(F_1 \wedge F_2)[x := t] & = & F_1[x := t] \wedge F_2[x := t] \\
(F_1 \vee F_2)[x := t] & = & F_1[x := t] \vee F_2[x := t] \\
(F_1 \Rightarrow F_2)[x := t] & = & F_1[x := t] \Rightarrow F_2[x := t] \\
(\forall x\ F_0)[x := t] & = & \forall x\ F_0
\end{array}
$$

Substitution dans une formule

définition inductive de la formule $F[x := t]$ lorsqu'aucune variable liée de F n'apparaît dans t

$$
\begin{array}{rcll}
\mathrm{true}[x := t] & = & \mathrm{true} & \\
\mathrm{false}[x := t] & = & \mathrm{false} & \\
p(t_1, \cdots, t_n)[x := t] & = & p(t_1[x := t], \cdots, t_n[x := t]) & (p \in \mathcal{P}_n) \\
(\neg F_0)[x := t] & = & \neg(F_0[x := t]) & \\
(F_1 \wedge F_2)[x := t] & = & F_1[x := t] \wedge F_2[x := t] & \\
(F_1 \vee F_2)[x := t] & = & F_1[x := t] \vee F_2[x := t] & \\
(F_1 \Rightarrow F_2)[x := t] & = & F_1[x := t] \Rightarrow F_2[x := t] & \\
(\forall x \ F_0)[x := t] & = & \forall x \ F_0 & \\
(\forall y \ F_0)[x := t] & = & \forall y \ (F_0[x := t]) & (x \neq y)
\end{array}
$$

Substitution dans une formule

définition inductive de la formule $F[x := t]$ lorsqu'aucune variable liée de F n'apparaît dans t

$$
\begin{array}{rcll}
\mathrm{true}[x := t] & = & \mathrm{true} & \\
\mathrm{false}[x := t] & = & \mathrm{false} & \\
p(t_1, \cdots, t_n)[x := t] & = & p(t_1[x := t], \cdots, t_n[x := t]) & (p \in \mathcal{P}_n) \\
(\neg F_0)[x := t] & = & \neg(F_0[x := t]) & \\
(F_1 \wedge F_2)[x := t] & = & F_1[x := t] \wedge F_2[x := t] & \\
(F_1 \vee F_2)[x := t] & = & F_1[x := t] \vee F_2[x := t] & \\
(F_1 \Rightarrow F_2)[x := t] & = & F_1[x := t] \Rightarrow F_2[x := t] & \\
(\forall x\ F_0)[x := t] & = & \forall x\ F_0 & \\
(\forall y\ F_0)[x := t] & = & \forall y\ (F_0[x := t]) & (x \neq y) \\
(\exists x\ F_0)[x := t] & = & \exists x\ F_0 & \\
(\exists y\ F_0)[x := t] & = & \exists y\ (F_0[x := t]) & (x \neq y)
\end{array}
$$

Substitution dans une formule

définition inductive de la formule $F[x := t]$ lorsqu'aucune variable liée de F n'apparaît dans t

- *exemple :* $((\exists y\ p(g(y, x))) \vee (\forall x\ p(h(x, z))))[x := f(k, y, z)]$

Substitution dans une formule

définition inductive de la formule $F[x := t]$ lorsqu'aucune variable liée de F n'apparaît dans t

- *exemple :* $((\exists y\ p(g(y, x))) \vee (\forall x\ p(h(x, z))))[x := f(k, y, z)]$
  1. renommage de la variable liée y qui apparaît dans $f(k, y, z)$

$$((\exists w\ p(g(w, x))) \vee (\forall x\ p(h(x, z))))[x := f(k, y, z)]$$

Substitution dans une formule

définition inductive de la formule $F[x := t]$ lorsqu'aucune variable liée de F n'apparaît dans t

- *exemple :* $((\exists y\ p(g(y, x))) \vee (\forall x\ p(h(x, z))))[x := f(k, y, z)]$
  1. renommage de la variable liée y qui apparaît dans $f(k, y, z)$

     $$((\exists w\ p(g(w, x))) \vee (\forall x\ p(h(x, z))))[x := f(k, y, z)]$$

  2. remplacement des occurrences libres de x par $f(k, y, z)$

     $$(\exists w\ p(g(w, f(k, y, z)))) \vee (\forall x\ p(h(x, z)))$$