Mathématiques pour l'informatique
Logique

Exercice 1. Quelle est la négation de la proposition suivante ?
$\forall x \in \boldsymbol{R}^{+*}, \forall \epsilon \in \boldsymbol{R}^{+*}, \exists \eta \in \boldsymbol{R}^{+*}, \forall y \in \boldsymbol{R}^{+*}, |y-x| < \eta \Longrightarrow |f(y)-f(x)| < \epsilon$

Exercice 2. Quelle est la négation de la proposition suivante ?
$\forall a \in \boldsymbol{N}^*, \forall b \in \boldsymbol{N}^*, \exists c \in \boldsymbol{N}, \exists d \in \boldsymbol{N}, \exists e \in \boldsymbol{N}, c = a \times d \wedge c = b \times e$
Cette proposition est-elle satisfaisable ? Tautologique ?

Pour ce qui suit, nous considérerons l'ensemble $\mathcal{N}$ de noms de variables et l'ensemble $\mathcal{F}$ de formules défini par induction structurelle comme :

$$\begin{array}{rcl} \mathcal{F} & ::= & \mathcal{N} \\ & | & \textsc{non}\ \mathcal{F} \\ & | & \mathcal{F}\ \textsc{implique}\ \mathcal{F} \end{array}$$

Rappelons que, si I est une fonction de $\mathcal{N}$ dans $\{V, F\}$, on peut définir à partir de I une *interprétation* J sur $\mathcal{F}$ par induction structurelle, de la manière suivante :

$$\begin{cases} \forall n \in \mathcal{N}, J(n) = I(n) \\ \forall f \in \mathcal{F}, J(\textsc{non}\ f) = \neg J(f) \\ \forall f, f' \in \mathcal{F}, J(f\ \textsc{implique}\ f') = f \Longrightarrow f' \end{cases}$$

Exercice 3. Si J est une interprétation construite comme précédemment à partir d'une fonction $I: \mathcal{N} \longrightarrow \{V, F\}$, prouver que, pour tout f dans $\mathcal{F}$, $J(f)$ est soit V soit F.

Exercice 4. Prouver que $\emptyset \models f\ \textsc{implique}\ f$ est *valide*.

Exercice 5. Prouver que $\emptyset \models f\ \textsc{implique non non}\ f$ est *valide*.

Exercice 6. * Quelle est la négation de la proposition suivante ?
$\forall a \in \boldsymbol{N}^*, \forall b \in \boldsymbol{N}^*, \exists c \in \boldsymbol{N}^*, \exists d \in \boldsymbol{N}, \exists e \in \boldsymbol{N}, a \times d \wedge b \times e$

Exercice 7. * Prouver que $\emptyset \models \textsc{non non}\ f\ \textsc{implique}\ f$ est valide.

Correction du TD 9

Exercice 8. Définir *strictement par induction* la soustraction de deux entiers de Peano. Implanter cela en OCaml. Prouver que la soustraction d'un entier à lui-même vaut toujours Zero.

Rappelons que les entiers de Peano sont les éléments de l'ensemble $\mathcal{N}$, défini par induction structurelle par

$$\begin{array}{rcl} \mathcal{N} & ::= & \text{Zero} \\ & | & S(\mathcal{N}) \end{array}$$

Tout comme pour l'addition ou la multiplication, la soustraction dans les entiers de Peano se définit comme une fonction soustraction: $\mathcal{N} \longrightarrow \mathcal{N} \longrightarrow \mathcal{N}$, de la manière suivante

$$\begin{cases} \text{si } x = \text{Zero, soustraction}(x) \text{ est la fonction définie par } \begin{array}{rcl} \mathcal{N} & \longrightarrow & \mathcal{N} \\ y & \longmapsto & \text{Zero} \end{array} \\ \text{si } x = \text{S(m), soustraction}(x) \text{ est la fonction définie par } \begin{array}{rcl} \mathcal{N} & \longrightarrow & \mathcal{N} \\ y & \longmapsto & \begin{cases} \text{si } y = \text{Zero, alors Zero} \\ \text{si } y = S(n) \text{, alors soustraction}(m)(n) \end{cases} \end{array} \end{cases}$$

En Ocaml, cela s'écrit

```
let rec soustraction = function
  | Zero      ->  (function _ -> 0)
  | S m as x ->  (function
```

```
      | Zero    -> x
      | S n     -> (soustraction m) n
  )
```

ou encore, de manière plus concise,

```
let rec soustraction x y = match (x,y) with
  | (Zero, _) -> Zero
  | (_, Zero) -> x
  | (S m, S n)-> soustraction m n
```

Prouvons maintenant que, pour tout entier de Peano p, soustraction$(p)(p)$ = Zero. Considérons l'hypothèse d'induction $\mathcal{H}_p$ « soustraction$(p)(p)$=Zero ».

Initialisation. Si $p=$Zero, soustraction(p) est la fonction qui à tout y associe Zero. En particulier, soustraction$(p)(p)=$Zero.

Induction. Plaçons-nous dans le cas où $p=S(m)$ et $\mathcal{H}_m$ est vraie.

Comme $p = S(m)$ et $p \neq$ Zero, par définition de soustraction, nous avons soustraction$(p)(p)$ = soustraction$(m)(m)$. Or, d'après l'hypothèse d'induction $\mathcal{H}_m$, nous avons soustraction$(m)(m)$ = Zero. Par conséquent, nous avons soustraction$(p)(p) = 0$, c'est-à-dire $\mathcal{H}_p$. Ce qui prouve le cas.

Nous venons de prouver par induction que $\forall p \in \mathcal{N}$, $\mathcal{H}_p$, c'est-à-dire que la soustraction d'un entier de Peano à lui-même est Zero.

Exercice 9. Définir l'opérateur XOR (« ou exclusif ») à partir des opérateurs $\wedge$et $\neg$.

Par définition, pour tout a et b, nous avons a XOR $b = (a \vee b) \wedge \neg(a \wedge b)$. Comme $a \vee b = \neg(\neg a \vee \neg b)$, nous pouvons réécrire a XOR $b = \neg(\neg a \wedge \neg b) \wedge \neg(a \wedge b)$.