5 Définitions récursives, Substitution

Exercice 5.1 *Récurrence sur les formules.*

1. On considère la fonction `simpl` définie par les équations suivantes dans lesquelles a est une variable propositionnelle choisie arbitrairement :

$$\begin{array}{llll}
\texttt{simpl}(\top) & = a \vee \neg a & \texttt{simpl}(P \wedge Q) & = \neg(\neg\texttt{simpl}(P) \vee \neg\texttt{simpl}(Q)) \\
\texttt{simpl}(\bot) & = \neg(a \vee \neg a) & \texttt{simpl}(P \vee Q) & = \texttt{simpl}(P) \vee \texttt{simpl}(Q) \\
\texttt{simpl}(p) & = p \quad p \text{ variable prop.} & \texttt{simpl}(P \Rightarrow Q) & = \neg\texttt{simpl}(P) \vee \texttt{simpl}(Q) \\
\texttt{simpl}(\neg P) & = \neg(\texttt{simpl}(P)) & &
\end{array}$$

   — Donner la formule résultat de $\texttt{simpl}(x \wedge y \Rightarrow z)$.
   — Que fait la fonction `simpl` en général (comment $\texttt{simpl}(P)$ est relié à P et quelles sont ses propriétés syntaxiques) ?

2. Donner les équations récursives qui définissent une fonction `ht` qui mesure la hauteur d'une formule (nombre maximal de connecteurs emboités)

$$\begin{array}{lll}
\texttt{ht}(P) & = \ldots. & \texttt{si } P \texttt{ atomique} \\
\texttt{ht}(\neg P) & = \ldots.. & \\
\texttt{ht}(P \circ Q) & = \ldots\ldots. & \circ \in \{\vee, \wedge, \Rightarrow\}
\end{array}$$

3. Démontrer par récurrence sur la hauteur d'une formule P que pour toute formule P, `simpl`(P) ne contient pas le symbole $\Rightarrow$.

4. Pour prouver qu'une propriété $\phi(P)$ est vérifiée pour toutes les formules propositionnelles P, on peut raisonner par récurrence **structurelle** sur les formules, suivant le principe :

   — Si on peut montrer que :
   (a) $\phi(P)$ est vérifiée lorsque P est une formule atomique en particulier $\phi(\top)$ et $\phi(\bot)$;
   (b) pour une formule propositionnelle A quelconque, en supposant que $\phi(A)$ est vérifiée, on peut montrer $\phi(\neg A)$
   (c) pour des formules propositionnelles A et B quelconques, en supposant que $\phi(A)$ et $\phi(B)$ sont vérifiées, on peut montrer $\phi(A \wedge B)$ ainsi que $\phi(A \vee B)$ et $\phi(A \Rightarrow B)$
   — Alors on peut en déduire que pour tout $P \in \textsc{Prop}$, $\phi(P)$ est vérifiée.

   Utiliser ce schéma pour montrer la propriété attendue de $\texttt{simpl}(P)$.

5. (optionnel) Justifier la correction de ce schéma de preuve en le ramenant à une récurrence sur les entiers.
   — On suppose que les conditions (1) à (3) du principe de récurrence sur les formules sont satisfaites par la propriété ϕ. Montrer par récurrence sur n que pour toute formule P telle que $\texttt{ht}(P) \leq n$, on a $\phi(P)$ est vérifiée.
   — En déduire que $\phi(P)$ est vérifiée pour toutes les formules P.

Exercice 5.2 *Définition récursive propositionnelle.*

1. Donner une formule équivalente à $\neg(P \Rightarrow Q)$ qui n'utilise que la conjonction et la négation.
2. On souhaite écrire une fonction qui pour toute formule propositionnelle P calcule une nouvelle formule $\texttt{neg}(P)$ qui est logiquement équivalente à $\neg P$ mais qui au lieu d'ajouter un symbole de négation en tête de la formule, effectue de possibles simplifications et n'ajoute éventuellement un nouveau symbole de négation qu'au niveau d'une formule atomique. Pour cela on utiliserale fait que les formules suivantes sont valides :

$$\neg\bot \Leftrightarrow \top \qquad \neg\neg A \Leftrightarrow A \qquad \neg(P \wedge Q) \Leftrightarrow \neg P \vee \neg Q \qquad \neg(P \vee Q) \Leftrightarrow \neg P \wedge \neg Q$$

(a) Calculer la valeur attendue pour $\texttt{neg}(A)$ avec A définie comme $(\neg P) \Rightarrow (Q \vee \neg(P \vee R))$. Pour quelle valeur de P, Q et R cette formule est-elle vraie ?

(b) Donner les équations récursives qui définissent $\texttt{neg}(P)$

(c) Montrer par récurrence structurelle sur la formule P que $\texttt{neg}(P)$ est logiquement équivalent à $\neg P$.

Exercice 5.3 *Sous-formules.* On dit qu'une formule Q est une sous-formule de P si $Q = P$ ou bien si la formule Q apparait sous un connecteur de P. C'est-à-dire $P = \neg P'$ et Q est une sous-formule de P' ou bien $P = P_1 \circ P_2$ et Q est une sous-formule de P_1 ou bien une sous-formule de P_2 avec $\circ$ un des 3 connecteurs binaires : $\{\vee, \wedge, \Rightarrow\}$.

1. Donner toutes les sous-formules de la formule $\neg(p \vee (q \wedge r)) \Rightarrow (p \wedge q)$
2. Donner les équations qui définissent la fonction $\texttt{sf}$ dans $\textsc{Prop} \to \wp(\textsc{Prop})$ qui à une formule propositionnelle P associe l'ensemble de ses sous-formules.
3. Trouver un majorant du nombre de sous-formules d'une formule P qui utilise n connecteurs logiques. Donner un exemple où ce majorant est atteint. Prouver ce résultat par récurrence structurelle sur la formule.
4. (optionnel) Même question pour un minorant du nombre de sous-formules.

Exercice 5.4 *Définition récursive sur les termes*

On considère un langage avec des symboles de fonctions `f` binaire, `g` unaire et une constante `a` ainsi qu'un symbole de prédicat `R` binaire.

Donner les équations récursives qui définissent une fonction `xin` qui étant donnée une formule P du calcul des prédicats sur le langage précédent, et une variable x, renvoie vrai si la variable x est libre dans la formule P et faux sinon.

Exercice 5.5 On considère un langage avec des symboles de fonctions `f` binaire, `g` unaire et une constante `a` ainsi qu'un symbole de prédicat `R` binaire.

Donner les formules résultats des substitutions suivantes :

1. $\texttt{R}(x, \texttt{g}(y))[x \leftarrow \texttt{f}(\texttt{a}, \texttt{g}(y)); y \leftarrow x]$
2. $\texttt{R}(x, \texttt{g}(y))[x \leftarrow \texttt{f}(\texttt{a}, \texttt{g}(y))][y \leftarrow x]$
3. $\forall x, \texttt{R}(x, \texttt{g}(y))[y \leftarrow \texttt{g}(z)]$
4. $\forall x, \texttt{R}(x, \texttt{g}(y))[y \leftarrow \texttt{g}(x)]$
5. $\forall x, \texttt{R}(x, \texttt{g}(y))[x \leftarrow z]$
6. $\forall x, \texttt{R}(x, \texttt{g}(y))[x \leftarrow y]$