Faculté des Sciences d'Orsay — Université Paris-Saclay
Licence Informatique, LDD Informatique, Mathématiques, semestre 5 — 2024–25
Eléments de logique pour l'informatique (Info 315) — 19 novembre 2024

Partiel - 23 octobre 2024

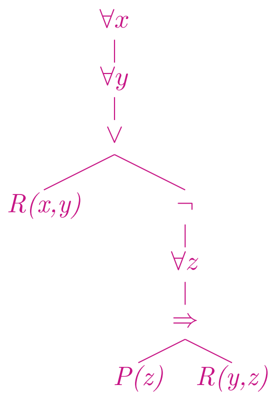
L'examen dure 2 heures. L'énoncé est composé de 7 pages et d'un QCM de 2 pages.
Toutes les réponses (sauf QCM) devront être clairement justifiées.
Le barème est indicatif. L'exercice 2 est à faire pour les licences mention informatique (parcours informatique et MIAGE) et la question 6 de l'exercice 3 est à faire pour les LDD Informatique, Mathématiques (dont magistère).
Le seul document autorisé est le polycopié de cours (document orange) non annoté.
Ne pas cacheter les copies !

Exercice 1 *QCM (5 points)*

Les nom et prénom ainsi que la filière doivent être reportés sur l'énoncé du QCM que vous rendrez avec votre copie (utiliser un stylo bleu ou noir pour cocher les cases). Sauf indication contraire, une bonne réponse rapporte $\frac{1}{4}$, une mauvaise réponse retire $\frac{1}{8}$ point, l'absence de réponse vaut 0 point.

Correction : *Voir correction individuelle des QCM.*

Concernant l'arbre sur la formule : les quantificateurs ont la portée la plus faible, ce qui veut dire qu'ils couvrent la formule qui suit la plus longue possible, ainsi $\forall x\, y, R(x,y) \lor \neg\forall z, P(z) \Rightarrow R(y,z)$ *se parenthèse en* $\forall x\, y, (R(x,y) \lor \neg\forall z, (P(z) \Rightarrow R(y,z)))$ *et l'arbre syntaxique est donc :*



Exercice 2 *Enigme,* ***uniquement pour les parcours Licence*** *(3 points)*

Trois amis Anne, Bob et Clara ont l'habitude de déjeûner ensemble. À la fin du repas ils prennent ou non un dessert.

Les habitudes des amis sont les suivantes :
— Si Anne prend un dessert, alors Bob en prend un aussi.
— Si Clara prend un dessert, alors Anne en prend un aussi.

De plus, on sait que ce jour-là, Anne ou Clara ont pris un dessert, mais que Bob et Clara n'ont pas tous les deux pris de dessert.

L'objectif est de déterminer qui a pris ou non un dessert ce jour-là. On introduit trois variables propositionnelles A (pour Anne), B (pour Bob) et C (pour Clara) qui sont vraies lorsque la personne correspondante a pris un dessert.

1. Traduire les hypothèses du problème en 4 formules propositionnelles.
2. Faire la table de vérité de ces 4 formules.
3. En déduire qui a pris un dessert et qui n'en a pas pris, justifier.

Correction :

1. *Les formules sont* $A \Rightarrow B, C \Rightarrow A, A \lor C, \neg(B \land C)$
   Pour la dernière phrase, celle-ci est la négation de "Bob et Clara ont tous les deux pris un dessert" ce qui est différent de l'affirmation "Ni Bob ni Clara n'ont pris de dessert" qui se traduirait par $\neg B \land \neg C$ *qui aboutit à un problème sans solution.*

2.

A	B	C	$A \Rightarrow B$	$C \Rightarrow A$	$A \vee C$	$\neg(B \wedge C)$
V	V	V	V	V	V	F
V	V	F	V	V	V	V
V	F	V	F	V	V	V
V	F	F	F	V	V	V
F	V	V	V	F	V	F
F	V	F	V	V	F	V
F	F	V	V	F	V	V
F	F	F	V	V	F	V

3. La seule situation qui rend vraies les 4 formules est la deuxième ligne : Anne et Bob ont pris un dessert mais pas Clara.

Exercice 3 *Tranformation de formules (Licence : 5 points, LDD : 8 points)* On considère ici des formules du calcul des prédicats, sur une signature quelconque qui contient au moins un symbole de prédicat P unaire et des variables propositionnelles X et Y. On rappelle ici la définition de la fonction qui compte le nombre de connecteurs logiques d'une formule.

$$\begin{array}{rll}
\texttt{nbc}(p) & = 0 & p \text{ atomique} \\
\texttt{nbc}(\neg A) & = 1 + \texttt{nbc}(A) & \\
\texttt{nbc}(A \circ B) & = 1 + \texttt{nbc}(A) + \texttt{nbc}(B) & o \in \{\vee, \wedge, \Rightarrow\} \\
\texttt{nbc}(\forall x, A) = \texttt{nbc}(\exists x, A) & = 1 + \texttt{nbc}(A) &
\end{array}$$

On souhaite tansformer une formule quelconque en une formule équivalente d'un point de vue logique mais qui ne contient que des formules atomiques (possiblement $\top$ et $\bot$), le connecteur de négation ($\neg$), la conjonction ($\wedge$) ainsi que le quantificateur universel ($\forall$). Une telle formule sera dite **réduite**.

1. En utilisant les lois de de Morgan (sans justification), donner des formules équivalentes à $X \vee Y$, $X \Rightarrow Y$ et $\exists x, P(x)$ qui sont réduites (n'utilisent que des formules atomiques et $\neg, \wedge, \forall$).
2. Définir par des équations récursives une fonction `transf` qui prend en argument une formule du calcul des prédicats générale et la transforme en une formule équivalente mais réduite.
3. Appliquer la transformation à la formule $(\forall x, P(x)) \Rightarrow \exists x, P(x)$ en suivant les étapes de votre définition.
4. On souhaite éviter l'apparition de formules $\neg\top$, $\neg\bot$ ou $\neg\neg A$ qui bien que réduites peuvent être simplifiées. Pour cela on utilisera deux fonctions `tropt` et `negopt`. Ces fonctions prennent en argument une formule du calcul des prédicats générale et la transforment en une formule en forme réduite. La fonction `tropt` transforme une formule A en une formule équivalente à A alors que `negopt` transforme une formule A en une formule équivalente à la négation de A ($\neg A$).
   Donner les équations récursives qui définissent `tropt` et `negopt` pour obtenir le résultat attendu.
5. On note A_1 la formule $X \vee X$ et A_2 la formule $(A_1 \wedge A_1) \vee (A_1 \wedge A_1)$. Quel est le nombre de connecteurs logiques dans les formules $\texttt{tropt}(A_1)$ et $\texttt{tropt}(A_2)$.
   Remarque Cette question peut être traitée à la main sur les exemples en cherchant des formules équivalentes réduites, même si les fonctions `tropt` et `transf` n'ont pas été définies.
6. **A faire uniquement par les LDD/Magistère**
   (a) Donner un majorant de $\texttt{nbc}(\texttt{tropt}(A))$ en fonction de $\texttt{nbc}(A)$.
   (b) Montrer que c'est un majorant par récurrence structurelle sur la formule A.
   (c) Proposer une formule de taille arbitrairement grande pour laquelle ce maximum est atteint.
   Remarque On pourra introduire la fonction $N(A) = \texttt{max}(\texttt{nbc}(\texttt{tropt}(A)), \texttt{nbc}(\texttt{negopt}(A)))$ (ou utiliser $N(A) = \texttt{nbc}(\texttt{transf}(A))$) et étudier les propriétés de $N(A)$ en fonction de la forme de la formule A.

Correction :

1.

$$X \vee Y \equiv \neg(\neg X \wedge \neg Y) \qquad X \Rightarrow Y \equiv \neg(X \wedge \neg Y) \qquad \exists x, P(x) \equiv \neg(\forall x, \neg P(x))$$

2.

$$\begin{array}{lll}
\texttt{transf}(p) & = p & p \text{ atomique} \\
\texttt{transf}(\neg A) & = \neg\,\texttt{transf}(A) & \\
\texttt{transf}(A \land B) & = \texttt{transf}(A) \land \texttt{transf}(B) & \\
\texttt{transf}(\forall x, A) & = \forall x, \texttt{transf}(A) & \\
\texttt{transf}(A \Rightarrow B) & = \neg(\texttt{transf}(A) \land \neg\,\texttt{transf}(B)) & \\
\texttt{transf}(A \lor B) & = \neg(\neg\,\texttt{transf}(A) \land \neg\,\texttt{transf}(B)) & \\
\texttt{transf}(\exists x, A) & = \neg\forall x, \neg\,\texttt{transf}(A) &
\end{array}$$

Les lois de de Morgan nous assurent que $\texttt{transf}(A) \equiv A$ *par ailleurs la fonction produit des formules qui ne contiennent que les connecteurs de négation, conjonction et quantification universelle.*

3. *Il s'agit ici de trouver la formule calculée par la définition précédente et non pas de donner une formule équivalente logiquement. Il est donc incorrect de simplifier la double-négation.*

$$\begin{array}{ll}
\texttt{transf}((\forall x, P(x)) \Rightarrow \exists x, P(x)) & = \neg(\texttt{transf}(\forall x, P(x)) \land \neg\,\texttt{transf}(\exists x, P(x))) \\
 & = \neg((\forall x, P(x)) \land \neg\neg\forall x, \neg P(x))
\end{array}$$

4. *On souhaite ici reprendre la définition de la transformation mais en ne gardant que les négations essentielles à savoir celles devant un symbole de prédicat, au dessus d'une conjonction ou d'un quantificateur universel. Cela se fait en utilisant deux fonctions mutuellement récursives, l'une pour transformer la forule l'autre pour transformer sa négation.*
On rappelle que lors de la définition d'une fonction par des équations récursives, il ne doit y avoir qu'une seule équation qui s'applique pour chaque formule et par ailleurs les seuls appels récursifs autorisés dans la partie droite portent sur des sous-formules strictes de l'argument dans la partie gauche (ces deux conditions permettent de garantir qu'il y a une unique valeur possible pour chaque formule passée en argument).

$$\begin{array}{lll}
\texttt{tropt}(p) & = p & p \text{ atomique} \\
\texttt{tropt}(\neg A) & = \texttt{negopt}(A) & \\
\texttt{tropt}(A \Rightarrow B) & = \neg(\texttt{tropt}(A) \land \texttt{negopt}(B)) & \\
\texttt{tropt}(A \lor B) & = \neg(\texttt{negopt}(A) \land \texttt{negopt}(B)) & \\
\texttt{tropt}(A \land B) & = \texttt{tropt}(A) \land \texttt{tropt}(B) & \\
\texttt{tropt}(\forall x, A) & = \forall x, \texttt{tropt}(A) & \\
\texttt{tropt}(\exists x, A) & = \neg\forall x, \texttt{negopt}(A) &
\end{array}$$

$$\begin{array}{lll}
\texttt{negopt}(\top) & = \bot & \\
\texttt{negopt}(\bot) & = \top & \\
\texttt{negopt}(p) & = \neg p & p \text{ atomique}, p \neq \bot, p \neq \top \\
\texttt{negopt}(\neg A) & = \texttt{tropt}(A) & \\
\texttt{negopt}(A \Rightarrow B) & = \texttt{tropt}(A) \land \texttt{negopt}(B) & \\
\texttt{negopt}(A \lor B) & = \texttt{negopt}(A) \land \texttt{negopt}(B) & \\
\texttt{negopt}(A \land B) & = \neg(\texttt{tropt}(A) \land \texttt{tropt}(B)) & \\
\texttt{negopt}(\forall x, A) & = \neg(\forall x, \texttt{tropt}(A)) & \\
\texttt{negopt}(\exists x, A) & = \forall x, \texttt{negopt}(A) &
\end{array}$$

5. *On a* $\texttt{tropt}(A_1) = \neg(\neg X \land \neg X)$ *qui a 4 connecteurs.*
On a $\texttt{tropt}(A_2) = \neg(\texttt{negopt}(A_1 \land A_1) \land \texttt{negopt}(A_1 \land A_1))$
et $\texttt{negopt}(A_1 \land A_1) = \neg(\texttt{tropt}(A_1) \land \texttt{tropt}(A_1))$*. Donc* $\texttt{negopt}(A_1 \land A_1)$ *a 10 connecteurs et* $\texttt{tropt}(A_2)$ *en a donc 22.*
On constate que sur cet exemple les fonctions `transf` *et* `tropt` *se comportent de manière identique. On a donc le même résultat.*

6. (a) *On voit que le cas le pire est celui de la disjonction pour* `tropt` *et de la conjonction pour* `negopt` *qui remplacent un connecteur par deux connecteurs.*
Sur l'exemple précédent on a A_1 *est de taille 1 et est transformée en une formule de taille 4 et* A_2 *est de taille 7 et transformée en une formule de taille 22.*
Si n *est le nombre de connecteurs logiques de* A *alors* $\texttt{tropt}(A)$ *et* $\texttt{negopt}(A)$ *ont au plus* $3n + 1$ *connecteurs logiques.*
Dans le cas de la fonction `transf`*, c'est le cas de la disjonction qui ajoute le plus de connecteurs (+3). Si on prend la formule* $A_0 = X$ *et* $A_{n+1} = A_n \lor X$ *on a* $\texttt{nbc}(\texttt{transf}(A_0)) = 0$ *et* $\texttt{nbc}(\texttt{transf}(A_{n+1})) = \texttt{nbc}(\texttt{transf}(A_n)) + 4$*. La borne est donc* $4n$*.*

   (b) *Soit* $\Phi(A)$ *la propriété à montrer :*
Si n *est le nombre de connecteurs logiques de* A*, alors* $\texttt{tropt}(A)$ *et* $\texttt{negopt}(A)$ *ont au plus* $3n + 1$ *connecteurs logiques.*
On fait la preuve par récurrence structurelle sur la formule A*.*

On note $N(A)$ le maximum de la taille de $\texttt{tropt}(A)$ et $\texttt{negopt}(A)$

On a $N(p) \leq 1$ si p est atomique, $N(A \circ B) \leq N(A) + N(B) + 2$, $N(\neg A) \leq N(A) + 1$, $N(\forall x, A) \leq N(A) + 2$ et $N(\exists x, A) \leq N(A) + 2$.

De là on établit que pour toute formule A on a $N(A) \leq 3|A| + 1$ avec $|A|$ la taille de A.

Preuve détaillée : *La propriété à montrer est pour toute formule A on a :*

$N(A) = \max(\texttt{nbc}(\texttt{tropt}(A)), \texttt{nbc}(\texttt{negopt}(A))) \leq 3\texttt{nbc}(A) + 1$

On regarde les différents cas pour la formule A.

- *p atomique : $\texttt{nbc}(\texttt{tropt}(p)) = \texttt{nbc}(p) = 0$ $\texttt{nbc}(\texttt{negopt}(p)) = \texttt{nbc}(\neg p) = 1$ et $\texttt{nbc}(p) = 0$, (définitions de $\texttt{tropt}$, $\texttt{negopt}$ et $\texttt{nbc}$)*
  on a bien $0 \leq 1 \leq 3 \times 0 + 1$
- *$\neg A$: $\texttt{nbc}(\texttt{tropt}(\neg A)) = \texttt{nbc}(\texttt{negopt}(A))$, $\texttt{nbc}(\texttt{negopt}(\neg A)) = \texttt{nbc}(\texttt{tropt}(A))$*
  et $\texttt{nbc}(\neg A) = 1 + \texttt{nbc}(A)$, (définitions de $\texttt{tropt}$, $\texttt{negopt}$ et $\texttt{nbc}$)
  Par hypothèse de récurrence, on a $\texttt{nbc}(\texttt{tropt}(A)) \leq 3 \times \texttt{nbc}(A) + 1$ et même chose pour $\texttt{nbc}(\texttt{negopt}(A))$
  Au final $\texttt{nbc}(\texttt{tropt}(\neg A)) = \texttt{nbc}(\texttt{negopt}(A)) \leq 3 \times \texttt{nbc}(A) + 1 \leq 3 \times (\texttt{nbc}(\neg A))$
- *$A \Rightarrow B$, $A \vee B$ et $A \wedge B$:*
  $\texttt{nbc}(\texttt{tropt}(A \Rightarrow B) = \texttt{nbc}(\neg(\texttt{tropt}(A) \wedge \texttt{negopt}(B))) = 2 + \texttt{nbc}(\texttt{tropt}(A)) + \texttt{nbc}(\texttt{negopt}(B))$
  $\texttt{nbc}(\texttt{tropt}(A \vee B)) = \texttt{nbc}(\neg(\texttt{negopt}(A) \wedge \texttt{negopt}(B)) = 2 + \texttt{nbc}(\texttt{negopt}(A)) + \texttt{nbc}(\texttt{negopt}(B))$
  $\texttt{nbc}(\texttt{tropt}(A \wedge B) = \texttt{nbc}(\texttt{tropt}(A) \wedge \texttt{tropt}(B))) = 1 + \texttt{nbc}(\texttt{tropt}(A)) + \texttt{nbc}(\texttt{tropt}(B))$
  Dans les trois cas ($\circ \in \{\vee, \wedge, \Rightarrow\}$), on a :
  $\texttt{nbc}(\texttt{tropt}(A \circ B)) \leq 2 + N(A) + N(B)$
  et $\texttt{nbc}(A \circ B) = 1 + \texttt{nbc}(A) + \texttt{nbc}(B)$, (définitions de $\texttt{tropt}$ et $\texttt{nbc}$)
  Par hypothèse de récurrence, on a $N(A) \leq 3 \times \texttt{nbc}(A) + 1$ et $N(B) \leq 3 \times \texttt{nbc}(B) + 1$
  Au final :
  $$\begin{aligned}\texttt{nbc}(\texttt{tropt}(A \circ B)) \leq 2 + N(A) + N(B) &\leq 2 + 3 \times \texttt{nbc}(A) + 1 + 3 \times \texttt{nbc}(B) + 1 \\ &= 3 \times \texttt{nbc}(A \circ B) + 1\end{aligned}.$$
  Le même raisonnement s'applique à $\texttt{nbc}(\texttt{negopt}(A \circ B))$
- *Dans le cas des quantificateurs $\forall x, A$ et $\exists y, A$ on a :*
  $\texttt{nbc}(\texttt{tropt}(\forall x, A)) = \texttt{nbc}(\forall x, \texttt{tropt}(A)) = 1 + \texttt{nbc}(\texttt{tropt}(A))$
  $\texttt{nbc}(\texttt{tropt}(\exists x, A)) = \texttt{nbc}(\neg \exists x, \texttt{negopt}(A)) = 2 + \texttt{nbc}(\texttt{negopt}(A))$
  Dans les deux cas, on a $\texttt{nbc}(\texttt{tropt}(\mathcal{Q}x, A)) \leq N(A) + 2$
  Par hypothèse de récurrence, on a $N(A) \leq 3 \times \texttt{nbc}(A) + 1$.
  Au final $\texttt{nbc}(\texttt{tropt}(\mathcal{Q}x, A)) \leq 2 + N(A) \leq 2 + 3 \times \texttt{nbc}(A) + 1 = 3 \times (\texttt{nbc}(A) + 1)$
  et donc $\texttt{nbc}(\texttt{tropt}(\mathcal{Q}x, A)) \leq 3 \times \texttt{nbc}(\mathcal{Q}x, A) + 1$.
  Le même raisonnement s'applique à $\texttt{nbc}(\texttt{negopt}(\mathcal{Q}x, A))$

On a couvert tous les cas et on a donc établi (par récurrence structurelle sur A) que la propriété $\texttt{nbc}(\texttt{tropt}(A)) \leq 3 \times \texttt{nbc}(A) + 1$ était vérifiée pour toutes les formules A.

(c) *Le cas le pire est atteint pour les formules A_i définies de la manière suivante :*

On définit la suite $A_1 \stackrel{\text{def}}{=} X \vee X$, et pour $i \geq 1$: $A_{2i} \stackrel{\text{def}}{=} A_{2i-1} \wedge A_{2i-1}$, $A_{2i+1} \stackrel{\text{def}}{=} A_{2i} \vee A_{2i}$

On a $\texttt{nbc}(A_1) = 1, \texttt{nbc}(A_{i+1}) = 2\texttt{nbc}(A_i) + 1$ Donc $\texttt{nbc}(A_i) = 2^i - 1$.

On a $\texttt{nbc}(\texttt{tropt}(A_1)) = 4$ et pour $i \geq 1$:

$\texttt{nbc}(\texttt{tropt}(A_{2i+1})) = \texttt{nbc}(\neg(\texttt{negopt}(A_{2i}) \wedge \texttt{negopt}(A_{2i}))) = 2\texttt{nbc}(\texttt{negopt}(A_{2i}) + 2$

$\texttt{nbc}(\texttt{negopt}(A_{2i})) = \texttt{nbc}(\neg(\texttt{tropt}(A_{2i-1}) \wedge \texttt{tropt}(A_{2i-1}))) = 2\texttt{nbc}(\texttt{tropt}(A_{2i-1}) + 2$

Si on considère la suite x_i avec $x_i = \texttt{nbc}(\texttt{tropt}(A_i))$ si i est impair et $x_i = \texttt{nbc}(\texttt{negopt}(A_i))$ si i est pair. On a

$x_1 = 4$ et $x_{i+1} = 2x_i + 2$ ce qui se résoud en

$x_{i+1} = 4 \times 2^i + 2 \times (2^i - 1) = 6 \times 2^i - 2 = 3 \times 2^{i+1} - 2 = 3 \times (2^{i+1} - 1) + 1$

On a donc bien $x_{i+1} = 3 \times \texttt{nbc}(A_{i+1}) + 1$ La formule A_i est donc un exemple de formule arbitrairement grande où la borne est atteinte.

Exercice 4 *Modélisation en logique du premier ordre (7 points)*
Le jeu de Nim consiste à se donner un paquet d'allumettes. A tour de rôle chaque joueur peut choisir de retirer 1, 2 ou 3 allumettes. Le joueur qui gagne est celui qui retire les dernières allumettes.

On souhaite modéliser un jeu à deux joueurs comme le jeu de Nim (ou encore les échecs, les dames) pour étudier en particulier les questions de stratégie.

On désignera par le terme "les blancs", le joueur qui joue en premier et par le terme "les noirs", le joueur qui joue en second.

Pour modéliser un jeu de cette nature, on va s'intéresser aux états successifs du jeu (par exemple la disposition des pièces sur un échiquier ou pour un jeu de Nim le nombre d'allumettes restantes, ainsi qu'un booléen représentant le joueur dont c'est le tour).

On utilise une signature avec les symboles suivants :

— $\mathtt{init}$: constante représentant l'état initial du jeu.

— $\mathtt{coupB}$, un symbole de prédicat binaire tel que $\mathtt{coupB}(s, s')$ est vrai lorsque s' est un état possible après un coup joué par les blancs sur un jeu à l'état s.
Par exemple pour le jeu de Nim, si un état est interprété comme le nombre d'allumettes restantes et un booléen représentant le joueur qui doit jouer, on aura $\mathtt{coupB}(s, s')$ vrai si et seulement dans l'état s, c'est au tour des blancs de jouer, si la différence entre le nombre d'allumettes de s et celui de s' est comprise entre 1 et 3 et si dans l'état s' c'est aux noirs de jouer.

— $\mathtt{coupN}$, un symbole de prédicat binaire comme le précédent mais pour les coups joués par les noirs.

— $\mathtt{B}$ un symbole de prédicat unaire tel que $\mathtt{B}(s)$ est vrai lorsque c'est aux blancs de jouer.

— $\mathtt{GaB}$ et $\mathtt{GaN}$, des symboles de prédicat unaires tels que $\mathtt{GaB}(s)$ (resp. $\mathtt{GaN}(s)$) est vrai lorsque l'état s est un état final gagnant pour les blancs (resp. les noirs).

— $\mathtt{StB}$ et $\mathtt{StN}$, des symboles de prédicat unaires tels que $\mathtt{StB}(s)$ (resp. $\mathtt{StN}(s)$) est vrai lorsque les blancs (resp. les noirs) ont une stratégie gagnante à partir de l'état s.

— le prédicat binaire d'égalité qui sera noté de manière infixe $t = s$.

1. Traduire en langue naturelle (en français ou en anglais correct) les formules suivantes.
Penser à bien positionner les parenthèses avant de démarrer la traduction, autant que possible on évitera de donner des noms aux variables liées, en utilisant plutôt des formulations impersonnelles.
   (a) $\forall s, \mathtt{GaB}(s) \Rightarrow \neg \mathtt{GaN}(s)$
   (b) $\forall s, \mathtt{GaB}(s) \Rightarrow \forall s', \neg \mathtt{coupN}(s, s')$
   (c) $\exists s, \forall s', \mathtt{coupN}(s, s') \Rightarrow \neg \mathtt{GaN}(s')$
   (d) $(\forall s, \mathtt{coupB}(\mathtt{init}, s) \Rightarrow \mathtt{StN}(s)) \Rightarrow \mathtt{StB}(\mathtt{init})$

   Correction : *Attention aux traductions ; Un état qui n'est pas gagnant pour un joueur ne peut pas se traduire en état perdant pour ce joueur. En effet, un état perdant pour un joueur est un état dans lequel son adversaire a gagné ou bien il ne peut plus jouer, un état peut être non-gagnant, simplement parce qu'il reste des coups à jouer sans préjuger si le joueur au final gagnera ou non.*
   (a) $\forall s, (GaB(s) \Rightarrow \neg GaN(s))$ *Un état final gagnant pour les blancs n'est pas gagnant pour les noirs*
   (b) $\forall s, (GaB(s) \Rightarrow \forall s', \neg coupN(s, s'))$ *A partir d'un état final gagnant pour les blancs, il n'y a pas de coup possible pour les noirs.*
   (c) $\exists s, \forall s', (coupN(s, s') \Rightarrow \neg GaN(s'))$ *Il existe un état à partir duquel tous les coups des noirs mènent à un état dans lequel ils n'ont pas gagné. Ou encore : Il existe un état à partir duquel il n'y a pas de coup gagnant pour les noirs.*
   (d) $(\forall s, (coupB(init, s) \Rightarrow StN(s))) \Rightarrow StB(init)$ *Si après n'importe quel coup initial joué par les blancs, les noirs ont une stratégie gagnante, alors les blancs ont également une stratégie gagnante pour le jeu initial.*
   Remarque *Cette propriété peut paraître étrange, elle est néanmoins prouvables pour certaines classes de jeux (en particulier des jeux symétriques et sans partie nulle) et elle sert à montrer (par l'absurde) qu'il n'y a pas de stratégie gagnante pour les noirs.*

2. Donner des formules logiques du premier ordre qui n'utilisent que les symboles de la signature donnée pour représenter les propositions suivantes.
Attention à bien respecter la syntaxe des formules de la logique. Pour les phrases complexes, penser à les décomposer en morceaux plus simples.
   (a) Il y a exactement deux coups possibles pour les blancs à partir de l'état initial.

(b) Un état dans lequel c'est aux blancs de jouer mais à partir duquel ils ne peuvent rien jouer (pas de coup possible) est gagnant pour les noirs.

(c) A partir du même état, il n'est pas possible de jouer à la fois un coup pour les blancs et un coup pour les noirs.

(d) Aucun état final gagnant pour les blancs n'a une stratégie gagnante pour les noirs

Correction :

(a) $\exists x\, y, \texttt{coupB}(\texttt{init}, x) \wedge \texttt{coupB}(\texttt{init}, y) \wedge x \neq y \wedge \forall z, \texttt{coupB}(\texttt{init}, z) \Rightarrow (z = x \vee z = y)$

(b) $\forall x, \texttt{B}(x) \wedge (\forall y, \neg\texttt{coupB}(x, y)) \Rightarrow \texttt{GaN}(x)$

(c) $\forall x\, y\, z, \texttt{coupB}(x, y) \Rightarrow \neg\texttt{coupN}(x, z)$ *ou de manière équivalente*
$\forall x, (\exists y\, \texttt{coupB}(x, y)) \Rightarrow \neg(\exists z\, \texttt{coupN}(x, z))$
ou encore $\forall x, \neg((\exists y\, \texttt{coupB}(x, y)) \wedge (\exists z\, \texttt{coupN}(x, z)))$ *La formule* $\forall x\, y, \texttt{coupB}(x, y) \Rightarrow \neg\texttt{coupN}(x, y)$ *n'est pas suffisante en effet elle dit que entre deux états donnés il ne peut pas y avoir à la fois un coup possible des blancs et un coup possible des noirs, mais il pourrait très bien y avoir un coup des blancs qui amène à un état et un coup des noirs qui amène à un autre état, ce qui contradirait la propriété demandée.*

(d) $\forall x, \texttt{GaB}(x) \Rightarrow \neg\texttt{StN}(x)$ *ou encore* $\neg\exists x, (\texttt{GaB}(x) \wedge \texttt{StN}(x))$

3. Donner une formule équivalente à la formule de la question 1d qui de plus soit en forme normale de négation.

Correction :

$$\begin{aligned}(\forall s, \texttt{coupB}(\texttt{init}, s) \Rightarrow \texttt{StN}(s)) \Rightarrow \texttt{StB}(\texttt{init}) \quad &\equiv \neg(\forall s, (\texttt{coupB}(\texttt{init}, s) \Rightarrow \texttt{StN}(s))) \vee \texttt{StB}(\texttt{init})\\ &\equiv (\exists s, \neg(\texttt{coupB}(\texttt{init}, s) \Rightarrow \texttt{StN}(s))) \vee \texttt{StB}(\texttt{init})\\ &\equiv (\exists s, \texttt{coupB}(\texttt{init}, s) \wedge \neg\texttt{StN}(s)) \vee \texttt{StB}(\texttt{init})\end{aligned}$$

4. On se donne une interprétation de la signature pour représenter le jeu de Nim. Comme indiqué précédemment, le domaine sera formé des couples (n, b) avec n un entier naturel et b un booléen. On notera nB avec n un entier, la valeur $(n, \texttt{vrai})$ qui correspond au cas où il reste n allumettes et c'est aux blancs de jouer et nN la valeur $(n, \texttt{faux})$ qui correspond au cas où il reste n allumettes et c'est aux noirs de jouer.

La situation est finale s'il n'y a plus d'allumettes ($n = 0$). Elle est gagnante pour les blancs si c'est aux noirs de jouer et dans le cas contraire, elle est gagnante pour les noirs. L'interprétation de `GaN` contient donc pour seul élément la valeur notée $0B$.

Un exemple de partie est $6B \longrightarrow_B 4N \longrightarrow_N 2B \longrightarrow_B 0N$ dans laquelle les blancs qui jouent le dernier coup gagnent.

On suppose dans la suite que l'on part d'un jeu avec 6 allumettes et que les blancs jouent en premier.

(a) Quelle est l'interprétation de la constante `init` et des symboles de prédicat `coupB`, `coupN`, `B` et `GaB`.

(b) Soit la formule $A(x)$ dans laquelle la variable x est libre définie par

$$\neg\texttt{B}(x) \wedge \forall y, \texttt{coupN}(x, y) \Rightarrow \exists z, \texttt{coupB}(y, z) \wedge \texttt{GaB}(z)$$

i. Exprimer en langue naturelle le sens de cette formule en fonction de x.

ii. Pour quelles valeurs de x, la formule $A(x)$ est-elle vraie ? Justifier.

iii. En déduire la valeur de vérité de la formule $\exists x, \texttt{coupB}(\texttt{init}, x) \wedge A(x)$

Correction :

(a) `init` *est interprété par l'élément du domaine* $6B$ *(6 allumettes et c'est au tour des blancs).*
`B` *est interprété par l'ensemble* $\{6B, 5B, 4B, 3B, 2B, 1B, 0B\}$ *(tous les états où c'est aux blancs de jouer)*
`coupB` *est interprété par l'ensemble de couples :*
$\{(6B, 5N), (6B, 4N), (6B, 3N), (5B, 4N), (5B, 3N), (5B, 2N), (4B, 3N), (4B, 2N), (4B, 1N), (3B, 2N), (3B, 1N), (3B, 0N), (2B, 1N), (2B, 0N), (1B, 0N)\}$
`coupN` *est interprété par l'ensemble de couples :*
$\{(5N, 4B), (5N, 3B), (5N, 2B), (4N, 3B), (4N, 2B), (4N, 1B), (3N, 2B), (3N, 1B), (3N, 0B), (2N, 1B), (2N, 0B), (1N, 0B)\}$
`GaB` *est interprété par l'unique élément* $\{0N\}$

(b) *i. le sens de la formule* $A(x)$ *est : dans l'état* x*, c'est au tour des noirs de jouer et quelque soit le coup joué par les noirs, les blancs gagnent en un coup ensuite.*

ii. *Pour que les blancs gagnent, il doivent arriver à la position* $0N$ *ce qui n'est possible que s'ils étaient dans l'un des états* $3B, 2B, 1B$*, on cherche donc des états à partir desquels les noirs doivent jouer mais ne peuvent faire rien d'autre que d'aller dans un de ces états. Ce n'est pas le cas de* $5N$ *(les noirs peuvent retirer une seule allumette pour se retrouver dans l'état* $4B$ *à partir duquel ils peuvent gagner). C'est le cas de* $4N$*. Ce n'est pas le cas de* $3N$*,* $2N$ *ou* $1N$ *à partir desquels les noirs peuvent gagner en un coup.*
Dans l'interprétation du jeu de Nim à 6 allumettes, la formule $A(x)$ *est vraie dans un environnement où* x *à la valeur* $4N$*, elle est également vraie si* x *à la valeur* $0N$ *car alors aucun coup n'est possible pour les noirs, ce qui rend la formule* $\forall y, \mathtt{coupN}(x, y) \Rightarrow \exists z, \mathtt{coupB}(y, z) \wedge \mathtt{GaB}(z)$ *vraie.*

iii. *La formule* $\exists x, \mathtt{coupB}(\mathtt{init}, x) \wedge A(x)$ *est vraie dans l'interprétation du jeu de Nim à 6 allumettes : on prend pour* x *la valeur* $4N$*, on a bien* $\mathtt{coupB}(\mathtt{init}, x)$ *qui est vrai et* $A(x)$ *qui est vrai.*