Exercice 1 : logique propositionnelle et résolution (*4 points*)

Pierre, Antoine et Thomas sont trois étudiants qui passent un examen. Après l'examen, Pierre dit : "Si au moins l'un de nous trois a triché, alors nous avons triché tous les trois.". Puis Antoine ajoute "Si Pierre a triché, alors Thomas a triché."

Question 1 : Exprimer par une formule de la logique des propositions les déclarations de Pierre et Antoine, en utilisant les propositions suivantes : p : "Pierre a triché.", t : "Thomas a triché.", a : "Antoine a triché.".

(2 points : 1 point par formule)
Pierre : $(p \vee t \vee a) \Rightarrow (p \wedge t \wedge a)$
Antoine : $p \Rightarrow t$

Question 2 : En supposant que Pierre et Antoine mentent toujours, que peut-on en déduire sur ceux qui ont triché et ceux qui n'ont pas triché, **en n'utilisant que la méthode de résolution** ? (l'usage de toute autre méthode que la méthode de résolution est hors sujet, donc ne rapporte aucun point)

(0,5 points) La négation de l'affirmation de Pierre est $\neg((p \vee t \vee a) \Rightarrow (p \wedge t \wedge a))$, qui est équivalente à $(p \vee t \vee a) \wedge \neg(p \wedge t \wedge a)$ et

$$(p \vee t \vee a) \wedge (\neg p \vee \neg t \vee \neg a).$$

(0,5 points) La négation de l'affirmation d'Antoine est $p \wedge \neg t$.
(0,5 points) Ceci donne les clauses $C_1 = p \vee t \vee a$, $C_2 = \neg p \vee \neg t \vee \neg a$, $C_3 = p$ et $C_4 = \neg t$.
(0,5 points) $C_5 = res(C_2, C_3) = \neg t \vee \neg a$, $C_6 = res(C_1, C_4) = p \vee a$, $C_7 = res(C_5, C_6) = \neg t \vee p$. Ces trois nouvelles clauses sont subsumées par p ou par $\neg t$.
(0,5 points) Aucune autre résolution n'est possible, donc, par résolution, on apprend que Pierre a triché et que Thomas n'a pas triché. On n'apprend rien sur Antoine.

Exercice 2 : preuve formelle en déduction naturelle (*3 points*)

On considère le système formel sans axiomes composé des 5 règles de déduction suivantes :

$$\Rightarrow_I : \frac{P \vdash Q}{P \Rightarrow Q} \qquad \Rightarrow_E : \frac{P, P \Rightarrow Q}{Q} \qquad \wedge_{EG} : \frac{P \wedge Q}{P} \qquad \wedge_{ED} : \frac{P \wedge Q}{Q} \qquad \wedge_I : \frac{P, Q}{P \wedge Q}$$

On admet que ce fragment du système de la déduction naturelle est correct et complet pour les formules propositionnelles construites uniquement à partir des connecteurs logiques $\Rightarrow$ et $\wedge$.

En n'utilisant que les règles de ce système, présenter sous forme d'arbres une démonstration formelle complète de la tautologie

$$((x \wedge y) \Rightarrow z) \Rightarrow (x \Rightarrow (y \Rightarrow z)). \tag{1}$$

On exige une présentation claire de cette démonstration, où chaque étape est détaillée. Chaque règle appliquée doit être citée et les fins de démonstration (feuilles des arbres) doivent indiquer correctement les hypothèses qu'elles utilisent. Chaque sous-preuve doit clairement indiquer ses hypothèses.

(3 points)

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[x],[y]}{x \wedge y}\ \wedge_I \quad [(x \wedge y) \Rightarrow z]}{y \vdash \quad z}\ \Rightarrow_E}{x \vdash \quad y \Rightarrow z}\ \Rightarrow_I}{(x \wedge y) \Rightarrow z \vdash \quad x \Rightarrow (y \Rightarrow z)}\ \Rightarrow_I}{((x \wedge y) \Rightarrow z) \Rightarrow (x \Rightarrow (y \Rightarrow z))}\ \Rightarrow_I$$

Exercice 3 : vérification de type (*2 points*)

On considère la restriction suivante du système des règles de vérification et d'inférence des types polymorphes du cours :

$$\frac{f\ :\ \alpha_1 \times \alpha_2 \to \beta,\ e_1\ :\ \alpha_1,\ e_2\ :\ \alpha_2}{f(e_1, e_2)\ :\ \beta}\quad (app) \qquad \text{et} \qquad \frac{x\ :\ \alpha\ \vdash\ e\ :\ \beta}{\lambda\, x\, .\, e\ :\ (\alpha \to \beta)}\quad (abs).$$

Utiliser ces règles pour démontrer formellement le théorème

$$f : \alpha_1 \times \alpha_2 \to \alpha_3 \vdash (\lambda\, x_1\ .\ (\lambda\, x_2\ .\ f(x_1, x_2)))\ :\ (\alpha_1 \to (\alpha_2 \to \alpha_3)).$$

(2 points) Démonstration de

$$f : \alpha_1 \times \alpha_2 \to \alpha_3 \vdash (\lambda\, x_1\ .\ (\lambda\, x_2\ .\ f(x_1, x_2)))\ :\ (\alpha_1 \to (\alpha_2 \to \alpha_3))\ :$$

$$\dfrac{\dfrac{\dfrac{[f : \alpha_1 \times \alpha_2 \to \alpha_3],\quad [x_1\ :\ \alpha_1],\quad [x_2\ :\ \alpha_2]}{f(x_1, x_2)\ :\ \alpha_3}\ (app)}{x_2\ :\ \alpha_2 \vdash \quad (\lambda\, x_2\ .\ f(x_1, x_2))\ :\ (\alpha_2 \to \alpha_3)}\ (abs)}{x_1\ :\ \alpha_1 \vdash \quad \lambda\, x_1\ .\ (\lambda\, x_2\ .\ f(x_1, x_2))\ :\ (\alpha_1 \to (\alpha_2 \to \alpha_3))}\ (abs)$$

$$f : \alpha_1 \times \alpha_2 \to \alpha_3 \vdash$$

Exercice 4 : Calculs et preuves sur les listes (*11 points*)

On rappelle que le type inductif $Liste(\alpha)$ des listes dont tous les éléments sont de type α est défini par les règles

$$\frac{}{Nil : Liste(\alpha)} \qquad \text{et} \qquad \frac{a : \alpha, l : Liste(\alpha)}{Cons(a, l) : Liste(\alpha)}.$$

Le constructeur *Nil* représente la liste vide. Le constructeur *Cons* représente l'ajout d'un élément en tête de liste.

On veut définir les fonctions suivantes :

— $first : Liste(\alpha) - \{Nil\} \to \alpha$, qui extrait l'élément en tête de toute liste non vide,

— *last*, qui extrait le dernier élément d'une liste non vide,

— $changeFirst : \alpha \times Liste(\alpha) \to Liste(\alpha)$, telle que $changeFirst(x, l)$ place l'élément x à la place du premier élément de la liste l, et

— *changeLast*, telle que $changeLast(x, l)$ place l'élément x à la place du dernier élément de la liste l.

Les fonctions *changeFirst* et *changeLast* doivent aussi être définies pour la liste vide. Dans ce cas, elles retournent la liste vide.

Par exemple,
$\mathit{first}(\mathit{Cons}(1, \mathit{Cons}(2, \mathit{Cons}(3, \mathit{Nil}))))$ vaut 1,
$\mathit{last}(\mathit{Cons}(1, \mathit{Cons}(2, \mathit{Cons}(3, \mathit{Nil}))))$ vaut 3,
$\mathit{changeFirst}(6, \mathit{Nil})$ vaut Nil,
$\mathit{changeFirst}(6, \mathit{Cons}(1, \mathit{Cons}(2, \mathit{Cons}(3, \mathit{Nil}))))$ vaut $\mathit{Cons}(6, \mathit{Cons}(2, \mathit{Cons}(3, \mathit{Nil})))$,
$\mathit{changeLast}(6, \mathit{Nil})$ vaut Nil et
$\mathit{changeLast}(6, \mathit{Cons}(1, \mathit{Cons}(2, \mathit{Cons}(3, \mathit{Nil}))))$ vaut $\mathit{Cons}(1, \mathit{Cons}(2, \mathit{Cons}(6, \mathit{Nil})))$.

Question 1 : Donner le type des fonctions *last* et *changeLast*.

(1 point) $\mathit{last} : \mathit{Liste}(\alpha) - \{\mathit{Nil}\} \to \alpha$

(1 point) $\mathit{changeLast} : \alpha \times \mathit{Liste}(\alpha) \to \mathit{Liste}(\alpha)$

Question 2 : Définir inductivement les fonctions *first*, *last*, *changeFirst* et *changeLast* par des égalités.

(4 points : 0,5 points par égalité. Pour $\mathit{last}(\mathit{Cons}(x, l))$, la condition $l \neq \mathit{Nil}$ est exigée)

$$\mathit{first}(\mathit{Cons}(x, l)) = x \quad (2)$$

$$\mathit{last}(\mathit{Cons}(x, \mathit{Nil})) = x \quad (3)$$
$$\mathit{last}(\mathit{Cons}(x, l)) = \mathit{last}(l) \text{ pour } l \neq \mathit{Nil} \quad (4)$$

$$\mathit{changeFirst}(x, \mathit{Nil}) = \mathit{Nil} \quad (5)$$
$$\mathit{changeFirst}(x, \mathit{Cons}(y, l)) = \mathit{Cons}(x, l) \quad (6)$$

$$\mathit{changeLast}(x, \mathit{Nil}) = \mathit{Nil} \quad (7)$$
$$\mathit{changeLast}(x, \mathit{Cons}(y, \mathit{Nil})) = \mathit{Cons}(x, \mathit{Nil}) \quad (8)$$
$$\mathit{changeLast}(x, \mathit{Cons}(y, l)) = \mathit{Cons}(y, \mathit{changeLast}(x, l)) \text{ pour } l \neq \mathit{Nil} \quad (9)$$

Question 3 : Démontrer inductivement que, pour toute liste non vide, on a
$\mathit{changeFirst}(\mathit{last}(l), \mathit{changeLast}(\mathit{first}(l), l)) = \mathit{changeLast}(\mathit{first}(l), \mathit{changeFirst}(\mathit{last}(l), l))$.
Détailler les étapes de cette démonstration formelle.

Démonstration inductive que, pour toute liste l non vide, on a

$$\mathit{changeFirst}(\mathit{last}(l), \mathit{changeLast}(\mathit{first}(l), l)) = \mathit{changeLast}(\mathit{first}(l), \mathit{changeFirst}(\mathit{last}(l), l))$$

(2 points) Cas de base : $l = \mathit{Cons}(x, \mathit{Nil})$.

$$\begin{array}{ll} & \mathit{changeFirst}(\mathit{last}(\mathit{Cons}(x, \mathit{Nil})), \mathit{changeLast}(\mathit{first}(\mathit{Cons}(x, \mathit{Nil})), \mathit{Cons}(x, \mathit{Nil}))) \\ = & \quad \text{par (3)} \\ & \mathit{changeFirst}(x, \mathit{changeLast}(\mathit{first}(\mathit{Cons}(x, \mathit{Nil})), \mathit{Cons}(x, \mathit{Nil}))) \\ = & \quad \text{par (2)} \\ & \mathit{changeFirst}(x, \mathit{changeLast}(x, \mathit{Cons}(x, \mathit{Nil}))) \\ = & \quad \text{par (8)} \\ & \mathit{changeFirst}(x, \mathit{Cons}(x, \mathit{Nil})) \\ = & \quad \text{par (6)} \\ & \mathit{Cons}(x, \mathit{Nil}) \end{array}$$

et

$$\begin{array}{ll} & \mathit{changeLast}(\mathit{first}(\mathit{Cons}(x, \mathit{Nil})), \mathit{changeFirst}(\mathit{last}(\mathit{Cons}(x, \mathit{Nil})), \mathit{Cons}(x, \mathit{Nil}))) \\ = & \quad \text{par (2)} \\ & \mathit{changeLast}(x, \mathit{changeFirst}(\mathit{last}(\mathit{Cons}(x, \mathit{Nil})), \mathit{Cons}(x, \mathit{Nil}))) \\ = & \quad \text{par (3)} \\ & \mathit{changeLast}(x, \mathit{changeFirst}(x, \mathit{Cons}(x, \mathit{Nil}))) \\ = & \quad \text{par (6)} \\ & \mathit{changeLast}(x, \mathit{Cons}(x, \mathit{Nil})) \\ = & \quad \text{par (8)} \\ & \mathit{Cons}(x, \mathit{Nil}) \end{array}$$

On constate l'égalité syntaxique des termes réduits et on en déduit l'égalité des termes initiaux.
(3 points) Induction : $l = \mathit{Cons}(x, m)$ avec $m \neq \mathit{Nil}$. On suppose la propriété vraie pour m.

$$\begin{array}{ll} & \mathit{changeFirst}(\mathit{last}(\mathit{Cons}(x, m)), \mathit{changeLast}(\mathit{first}(\mathit{Cons}(x, m)), \mathit{Cons}(x, m))) \\ = & \quad \text{par (4)} \\ & \mathit{changeFirst}(\mathit{last}(m), \mathit{changeLast}(\mathit{first}(\mathit{Cons}(x, m)), \mathit{Cons}(x, m))) \\ = & \quad \text{par (2)} \\ & \mathit{changeFirst}(\mathit{last}(m), \mathit{changeLast}(x, \mathit{Cons}(x, m))) \\ = & \quad \text{par (9)} \\ & \mathit{changeFirst}(\mathit{last}(m), \mathit{Cons}(x, \mathit{changeLast}(x, m))) \\ = & \quad \text{par (6)} \\ & \mathit{Cons}(\mathit{last}(m), \mathit{changeLast}(x, m)) \end{array}$$

et

$$\begin{array}{ll} & \mathit{changeLast}(\mathit{first}(\mathit{Cons}(x, m)), \mathit{changeFirst}(\mathit{last}(\mathit{Cons}(x, m)), \mathit{Cons}(x, m))) \\ = & \quad \text{par (2)} \\ & \mathit{changeLast}(x, \mathit{changeFirst}(\mathit{last}(\mathit{Cons}(x, m)), \mathit{Cons}(x, m))) \\ = & \quad \text{par (4)} \\ & \mathit{changeLast}(x, \mathit{changeFirst}(\mathit{last}(m), \mathit{Cons}(x, m))) \\ = & \quad \text{par (6)} \\ & \mathit{changeLast}(x, \mathit{Cons}(\mathit{last}(m), m)) \\ = & \quad \text{par (9)} \\ & \mathit{Cons}(\mathit{last}(m), \mathit{changeLast}(x, m)) \end{array}$$

On constate l'égalité syntaxique des termes réduits et on en déduit l'égalité des termes initiaux. On peut remarquer qu'on n'a pas utilisé l'hypothèse d'induction.