Faculté des Sciences d'Orsay | Université Paris-Saclay
Licence Informatique, LDD Informatique, Mathématiques, semestre 5 | 2022-23
Eléments de logique pour l'informatique (Info 315) | 13 février 2024

Examen - 20 décembre 2023

L'examen dure 2 heures. L'énoncé est composé de 8 pages, le barème est indicatif.
Toutes les réponses (sauf QCM) devront être clairement justifiées.
Le seul document autorisé est une **page A4 manuscripte recto simple**.
Le tableau résumant les règles du système G est donné à la fin du sujet.
Recopiez le numéro d'anonymat sur les intercalaires et sur le QCM.
Correction :

Exercice 1 *QCM (4 points)* Le **numéro d'anonymat** de la copie principale (pas le numéro d'étudiant) doit être reporté sur l'énoncé du QCM. Utiliser un stylo bleu ou noir pour cocher les cases (pas de crayon papier). En cas d'erreur, indiquer en toute lettre la réponse souhaitée (vrai, faux ou pas de réponse).

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.

N'oubliez pas de rendre le QCM avec vos copies.

Correction : *Voir correction individuelle des QCM.*

Exercice 2 *Résolution (3 points)*
La signature contient un symbole de fonction unaire g et un symbole de prédicat unaire Q.

Par la méthode de résolution dont on détaillera les étapes, montrer que la formule ci-dessous est valide.

$$(\exists x, \neg Q(x) \land Q(g(g(x)))) \Rightarrow (\exists x, \neg Q(x) \land Q(g(x)))$$

Correction : *Soit la formule* $A \stackrel{\text{def}}{=} (\exists x, \neg Q(x) \land Q(g(g(x)))) \Rightarrow (\exists x, \neg Q(x) \land Q(g(x)))$

- *On calcule la forme clausale de la formule* $\neg A$
  - *forme normale de négation de* $\neg A$ *:*

    $$(\exists x, \neg Q(x) \land Q(g(g(x)))) \land (\forall x, Q(x) \lor \neg Q(g(x)))$$

  - *Suppression du quantificateur existentiel dans* $\exists x, \neg Q(x) \land Q(g(g(x)))$*, la formule n'ayant pas de variable libre, on introduit une nouvelle constante* a*. La formule* $\neg A$ *devient*

    $$\neg Q(a) \land Q(g(g(a))) \land \forall x, (Q(x) \lor \neg Q(g(x)))$$

  - *Mise en forme prénexe en faisant attention à conserver le bon parenthésage :*

    $$\forall x, \neg Q(a) \land Q(g(g(a))) \land (Q(x) \lor \neg Q(g(x)))$$

  - *La formule obtenue est en forme normale conjonctive, on obtient trois clauses*

  $C_1 \stackrel{\text{def}}{=} \neg Q(a) \qquad C_2 \stackrel{\text{def}}{=} Q(g(g(a))) \qquad C_3 \stackrel{\text{def}}{=} Q(x) \lor \neg Q(g(x))$
- *Etapes de résolution :*

$$\dfrac{[x \leftarrow a]\ \dfrac{\neg Q(a) \qquad Q(x) \lor \neg Q(g(x))}{\neg Q(g(a))} \qquad \dfrac{Q(g(g(a))) \qquad Q(x) \lor \neg Q(g(x))}{Q(g(a))}\ [x \leftarrow g(a)]}{\bot}$$

Exercice 3 *Preuve dans le système G (3 points)*
Soient X, Y des variables propositionnelles, utiliser le système G pour montrer que l'ensemble de formules $\{X \vee Y, X \Rightarrow Y, Y \Rightarrow X, \neg X \vee \neg Y\}$ est insatisfiable.
Correction : *Le calcul des séquents montre la validité d'un séquent, mais l'insatisfiabilité d'un ensemble de formules Γ est équivalent à la validité du séquent $\Gamma \vdash \bot$ qui s'écrit aussi $\Gamma \vdash$ sans formule à droite. On pouvait aussi raisonner en passant par la négation. L'insatisfiabilité de Γ est l'insatisfiabilité de la conjonction des formules qui est équivalente à la validité de la négation de cette conjonction qui est la disjonction des négations des formules. Si $\Gamma = \{A_1, \ldots, A_n\}$ on est ramené à prouver le séquent $\vdash \neg A_1, \ldots, \neg A_n$. On constate que l'application n fois de la règle droite pour la négation nous ramène au séquent précédent $A_1, \ldots, A_n \vdash$. Par contre le fait que le séquent $\vdash A_1, \ldots, A_n$ soit ou non prouvable ne dit rien sur l'insatisfiabilité de Γ.*

On fait donc une preuve dans le système G du séquent $X \vee Y, X \Rightarrow Y, Y \Rightarrow X, \neg X \vee \neg Y \vdash$

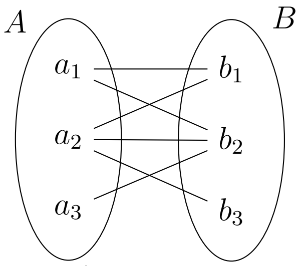
$$
\begin{prooftree}
\AxiomC{}
\LeftLabel{H}
\UnaryInfC{$X, X \Rightarrow Y, Y \Rightarrow X \vdash X$}
\LeftLabel{$\neg g$}
\UnaryInfC{$X, X \Rightarrow Y, Y \Rightarrow X, \neg X \vdash$}
\AxiomC{}
\LeftLabel{H}
\UnaryInfC{$X, Y, Y \Rightarrow X \vdash Y$}
\AxiomC{}
\LeftLabel{H}
\UnaryInfC{$X, Y \Rightarrow X \vdash X, Y$}
\LeftLabel{$\Rightarrow g$}
\BinaryInfC{$X, X \Rightarrow Y, Y \Rightarrow X \vdash Y$}
\LeftLabel{$\neg g$}
\UnaryInfC{$X, X \Rightarrow Y, Y \Rightarrow X, \neg Y \vdash$}
\LeftLabel{$\vee g$}
\BinaryInfC{$X, X \Rightarrow Y, Y \Rightarrow X, \neg X \vee \neg Y \vdash$}
\AxiomC{\textit{symétrique en XY}}
\UnaryInfC{$Y, X \Rightarrow Y, Y \Rightarrow X, \neg X \vee \neg Y \vdash$}
\LeftLabel{$\vee g$}
\BinaryInfC{$X \vee Y, X \Rightarrow Y, Y \Rightarrow X, \neg X \vee \neg Y \vdash$}
\end{prooftree}
$$

Exercice 4 *Modélisations (7 points).*

On cherche à modéliser une problème de couplage dans un graphe. On se donne deux ensembles distincts A et B finis de même taille et un ensemble d'arêtes E qui vont de A à B c'est-à-dire telles que si $(x, y) \in E$ alors $x \in A$ et $y \in B$.

Un couplage est donné par un sous-ensemble C d'arêtes tel que tout sommet appartient à exactement une arête de C. On supposera que dans le graphe initial tout sommet appartient à au moins une arête (sinon il n'existe pas de couplage).

Exemple. Soit le graphe :



Un couplage est donné par l'ensemble de 3 arêtes :

$$(a_1, b_1), (a_2, b_3), (a_3, b_2)$$

.

Partie 1 : modélisation au premier ordre (3 points)
Pour modéliser ce problème en logique du premier ordre, on se place dans un univers qui représente les sommets. On se donne trois prédicats binaires : $E(x, y)$ représente une arête du graphe, $C(x, y)$ représente une arête d'un couplage du graphe et l'égalité $x = y$.

Dans ce langage on peut exprimer le fait que x est un sommet de l'ensemble de départ du graphe par la formule $\exists y, E(x, y)$ et de manière analogue pour l'ensemble d'arrivée.

1. Donner une interprétation de ce langage correspondant au graphe de l'exemple : on explicitera le domaine et les interprétations de E et de C, l'interprétation de l'égalité étant donnée par l'égalité sur le domaine.
   Correction : *Le domaine est l'ensemble des sommets $\{a_1, a_2, a_3, b_1, b_2, b_3\}$.*
   L'interprétation du prédicat E est l'ensemble des couples de sommets correspondant aux

6 arètes : $\{(a_1, b_1), (a_1, b_2), (a_2, b_1), (a_2, b_2), (a_2, b_3), (a_3, b_2)\}$.
L'interprétation du prédicat C *est l'ensemble des couples de sommets correspondant aux 3 arètes du couplage :* $\{(a_1, b_1), (a_2, b_3), (a_3, b_2)\}$.

2. Exprimer par des formules de la logique du premier ordre sur la signature donnée les propriétés suivantes du couplage pour un graphe quelconque :
   (a) Les arêtes du couplage sont des arêtes du graphe initial.
   (b) Pour tout sommet de départ, il existe une arête qui part de ce sommet et qui est dans le couplage et pour tout sommet d'arrivée, il existe une arête qui arrive sur ce sommet et qui est dans le couplage.
   (c) Il y a au plus une arête du couplage qui part d'un sommet donné et il y a au plus une arête du couplage qui arrive sur un sommet donné.

   Correction :

   (a) $\forall x\, y, C(x, y) \Rightarrow E(x, y)$

   (b) Il fallait faire attention à restreindre les quantifications aux sommets de départ ou d'arrivée. Une manière concise de le faire est de partir de deux sommets x *et* y *liés par une arète, alors on sait que* x *est dans l'ensemble de départ et* y *dans celui d'arrivée. On obtient la formule* $\forall x\, y, E(x, y) \Rightarrow (\exists z, C(x, z)) \wedge (\exists z, C(z, y))$*. On pouvait aussi alternativement écrire* $\forall x, ((\exists y, E(x, y)) \Rightarrow \exists z, C(x, z)) \wedge ((\exists y, E(y, x)) \Rightarrow \exists z, C(z, x))$ *en faisant attention à parenthéser l'existentielle à gauche de la flèche (sinon on obtient une formule du genre paradoxe du buveur* $\exists x, P(x) \Rightarrow Q$ *qui se parenthèse* $\exists x, (P(x) \Rightarrow Q)$ *et qui est vraie dès qu'il existe un élement qui ne vérifie pas* $P(x)\ldots$*)*

   (c) $\forall x\, y\, C(x, y) \Rightarrow \forall z, (C(x, z) \Rightarrow y = z) \wedge (C(z, y) \Rightarrow x = z)$

Partie 2 : modélisation propositionnelle (4 points)
Pour modéliser ce problème de manière propositionnelle, on choisit de numéroter les sommets de A et de B de 1 jusqu'à n ($A = \{a_1, \ldots, a_n\}$ et $B = \{b_1, \ldots, b_n\}$) et on introduit autant de variables propositionnelles X_{ij} que d'arêtes. On a que X_{ij} est une variable du problème si et seulement si $(a_i, b_j) \in E$.

Dans la suite, on cherche à construire les contraintes du problème de couplage en faisant que la variable X_{ij} soit vraie lorsque l'arête correspondante est un élément du couplage entre A et B. Vous devez fournir des ensembles de formules propositionnelles qui ne contiennent donc que les variables propositionnelles X_{ij} correspondant au graphe et les connecteurs logiques $\neg, \wedge, \vee, \Rightarrow$.

1. Pour le graphe donné en exemple, avec donc les variables $X_{11}, X_{12}, X_{21}, X_{22}, X_{23}, X_{32}$:
   (a) donner un ensemble de clauses qui force chaque sommet (de A et de B) à apparaître sur au moins une arête du couplage ; par exemple on aura soit X_{11} soit X_{12} vrais.
   (b) donner un ensemble de clauses qui interdit à deux arêtes différentes du couplage de partager un sommet (de A ou de B) ; par exemple il n'est pas possible d'avoir à la fois X_{11} et X_{12} vrais.

   Correction :

   (a) On exprime qu'il y a au moins une arête du couplage qui part de chaque sommet de A *et qu'il y a au moins une arête qui arrive sur chaque sommet de* B*, on obtient 6 clauses :* $X_{11} \vee X_{12},\ X_{21} \vee X_{22} \vee X_{23},\ X_{32},\ \ X_{11} \vee X_{21},\ X_{32} \vee X_{22} \vee X_{12},\ X_{23}$

   (b) Lorsque d'un sommet partent ou arrivent plusieurs arètes, on exprime qu'une seule peut être dans le couplage, donc une arète dans le couplage interdit à toute autre

arête ayant le même départ ou la même arrivée à être dans le couplage. Par exemple si X_{11} est vraie alors X_{12} est fausse et X_{21} est fausse. Sous forme d'implication cela donne $X_{11} \Rightarrow \neg X_{12}$ et $X_{11} \Rightarrow \neg X_{21}$. On remarque qu'il est suffisant de regarder les paires d'arêtes non ordonnées, en effet la formule pour l'arête X_{12} serait $X_{12} \Rightarrow \neg X_{11}$, mais c'est la contraposée de $X_{11} \Rightarrow \neg X_{12}$ donc elle est équivalente et il est inutile de l'ajouter.

On exprime sous forme de clauses les propriétés : $X_{11} \Rightarrow \neg X_{12}$ devient $\neg X_{11} \vee \neg X_{12}$ Ce qui donne au final l'ensemble de clauses suivant :
$\neg X_{11} \vee \neg X_{12}$, $\neg X_{21} \vee \neg X_{22}$, $\neg X_{21} \vee \neg X_{23}$, $\neg X_{22} \vee \neg X_{23}$,
$\neg X_{11} \vee \neg X_{21}$, $\neg X_{32} \vee \neg X_{12}$, $\neg X_{32} \vee \neg X_{22}$, $\neg X_{22} \vee \neg X_{12}$

2. On se place dans le cas général d'un graphe avec deux ensembles disjoints de n sommets A et B et des arêtes de A vers B. On se donne l'ensemble X_{ij} de variables propositionnelles avec une variable par arête du graphe. On notera F l'ensemble des couples d'indices correspondant aux arêtes, c'est à dire que $(i, j) \in F$ si et seulement si le graphe possède une arête de a_i à b_j.

   Pour exprimer les formules propositionnelles demandées, on pourra utiliser des notations mathématiques, en particulier des notations ensemblistes en faisant varier les indices i et j et en se référant à l'ensemble F (mais le résultat doit correspondre à des formules propositionnelles sur les variables X_{ij}).

   (a) Donner un ensemble de clauses qui force chaque sommet (de A et de B) à apparaître sur une arête du couplage.

   (b) Donner un ensemble de clauses qui interdit à deux arêtes différentes du couplage de partager un sommet (de A ou de B).

   (c) Combien de clauses obtient-on au maximum en fonction du nombre de sommets n, en considérant qu'il peut y avoir au plus n arêtes qui arrivent ou partent d'un sommet.

   (d) A quelle condition (nécessaire et suffisante) sur l'ensemble de formules ainsi obtenu existe-t-il un couplage dans le graphe ?

Bonus Certaines de ces clauses sont-elles redondantes ?

Correction :

(a) On généralise le cas particulier précédent. Pour chaque sommet de départ i, on construit une clause qui est la disjonction des variables X_{ij} pour toutes les arêtes qui partent de i donc tous les j tels que $(i, j) \in F$. On obtient donc :

$$\{ \bigvee_{\{j|(i,j)\in F\}} X_{ij} | i = 1 \dots n\} \cup \{ \bigvee_{\{i|(i,j)\in F\}} X_{ij} | j = 1 \dots n\}$$

(b) Pour chaque paire d'arêtes de même départ $X_{ij}X_{ik}$ on a une clause $\neg X_{ij} \vee \neg X_{ik}$, de même pour les arêtes qui arrivent sur le même sommet. On obtient donc

$$\{\neg X_{ij} \vee \neg X_{ik} | (i, j) \in F, (i, k) \in F, j < k\} \cup \{\neg X_{ji} \vee \neg X_{ki} | (j, i) \in F, (k, i) \in F, j < k\}$$

3. Pour le premier ensemble de clauses il y en a exactement $2n$, pour le second, à chaque sommet il y a au plus n arètes donc $n(n-1)/2$ paires d'arêtes, on fait cela pour chaque sommet de départ et d'arrivée donc on a $n^2(n-1) + 2n$ clauses au maximum.

4. *Le graphe admet un couplage si et seulement si l'ensemble des clauses données est satisfiable, chaque modèle fournira un couplage différent.*
5. *Les clauses sont redondantes en effet si on considère juste les clauses d'existence et d'unicité sur les éléments de départ, cela nous dit qu'une solution X_{ij} au problème définit une application (totale) des sommets de départ vers les sommets d'arrivée. Il faut maintenant mettre des contraintes pour avoir une bijection. La seconde partie des clauses d'existence (en considérant les sommets de l'ensemble d'arrivée) nous donne la surjectivité (tout sommet d'arrivée a un antécédent) tandis que la seconde partie des clauses d'unicité nous donne l'injectivité (il n'y a pas deux arètes qui arrivent sur le même élément. Comme les deux ensembles de départ et d'arrivée ont le même nombre d'éléments, toute solution à l'un de ces ensembles de contraintes vérifiera aussi l'autre. Elles sont donc redondantes.*

Exercice 5 *Enigme (Licence, 3 points)*

Exercice à faire uniquement par les étudiants de Licence informatique (dont parcours Miage) et LDD MNSI les autres étudiants (LDD Informatique, Mathématiques et Magistère) traiteront à la place l'exercice 6.

Une île est habitée par deux catégories de personnes : les purs et les pires. Les purs disent toujours la vérité alors que les pires mentent toujours.

On introduit des variables propositionnelles pour chaque habitant x de l'Ile : $\texttt{pur}_x$ telles que $\texttt{pur}_x$ est vrai si x est pur et faux si x est pire.

Pour toute formule propositionnelle P, et tout habitant x, on introduit une variable propositionnelle $\texttt{dit}_{x,P}$, telle que $\texttt{dit}_{x,P}$ représente le fait que x affirme la formule P. Il y a a priori un nombre infini de variables propositionnelles mais dans les questions posées, on en utilisera qu'un nombre fini.

Toutes les affirmations devront être clairement justifiées par la méthode de votre choix.

1. Donner les formules (paramétrées par un individu x quelconque et une formule propositionnelle P quelconque) qui représentent le fait que un habitant pur dit la vérité et qu'un habitant pire ment.
2. Montrez que si $\texttt{a}$ et $\texttt{b}$ sont deux habitants de l'île et que $\texttt{a}$ dit que $\texttt{b}$ est pire, alors au moins l'un des deux est pire.
3. Montrez que si $\texttt{a}$ dit que $\texttt{a}$ et $\texttt{b}$ sont tous les deux pires, alors $\texttt{a}$ est pire.

Correction :

1. *Si x dit P et que x est pur alors P est vrai, ceci s'exprime par la formule : $\texttt{dit}_{P,x} \Rightarrow \texttt{pur}_x \Rightarrow P$*
   De même si x dit P et que x est pire alors P est faux, ceci s'exprime par la formule : $\texttt{dit}_{P,x} \Rightarrow \neg\texttt{pur}_x \Rightarrow \neg P$
2. *On peut raisonner par cas.*
   — *si a est pire alors l'un des deux est pire (a, peut-être b aussi)*
   — *si a est pur, comme a dit que b est pire, il dit la vérité et l'un des deux (b) est pire.*
   Plus simplement on peut repartir de notre formule $\texttt{dit}_{P,x} \Rightarrow \texttt{pur}_x \Rightarrow P$ avec pour x l'habitant a et pour formule P la formule $\neg\texttt{pur}_b$. L'énoncé nous dit que $\texttt{dit}_{P,x}$, on en déduit $\texttt{pur}_x \Rightarrow P$, c'est-à-dire $\texttt{pur}_a \Rightarrow \neg\texttt{pur}_b$ qui est équivalent à $\neg\texttt{pur}_a \vee \neg\texttt{pur}_b$ qui est ce qu'on doit justifier.
3. *On peut faire de même avec maintenant pour formule P la formule $\neg\texttt{pur}_a \wedge \neg\texttt{pur}_b$. On obtient $\neg\texttt{pur}_a \vee (\neg\texttt{pur}_a \wedge \neg\texttt{pur}_b)$ dont on déduit smplement que $\neg\texttt{pur}_a$ est toujours vrai.*
   On peut aussi raisonner informellement :

— *si a est pur, il dit la vérité et comme il dit que a est pire, il y a une contradiction. Donc a est pire. Comme a ment on en déduit que b est pur.*
— *si a est pur, il dit la vérité donc il ne peut pas dire faux. Donc a est pire.*

Exercice 6 *Satisfiabilité propositionnelle (LDD, 3 points)*

Exercice à faire uniquement par les étudiants de LDD Informatique, Mathématiques et Magistère à la place de l'exercice 5.

On s'intéresse à des clauses propositionnelles, c'est-à-dire des disjonctions de littéraux. On rappelle qu'un littéral l est une variable propositionnelle x ou bien la négation d'une variable propositionnelle $\neg x$.

Une clause est dite simplifiée si elle ne comporte pas deux littéraux portant sur la même variable (toujours possible si la clause n'est pas triviale).

On va montrer que tout ensemble de clauses simplifiées $\mathcal{E}$ peut être transformé en un ensemble de clauses ayant au plus trois littéraux.

La procédure de transformation de $\mathcal{E}$ est la suivante :

— On initialise un ensemble de clauses $\mathcal{C}$ avec l'ensemble $\mathcal{E}$
— Tant que $\mathcal{C}$ contient une clause de 4 littéraux ou plus, soit $l_1 \vee l_2 \vee l_3 \vee C$ une telle clause avec l_1, l_2, l_3 des littéraux et C une clause non vide. On introduit une nouvelle variable p et on remplace dans $\mathcal{C}$ la clause $l_1 \vee l_2 \vee l_3 \vee C$ par les deux clauses $l_1 \vee l_2 \vee p$ et $\neg p \vee l_3 \vee C$

1. Justifier pourquoi cette procédure termine (c'est-à-dire pourquoi au final $\mathcal{C}$ ne contiendra que des clauses de 3 littéraux au plus).

   Correction : *L'idée intuitive est que chaque étape "diminue" la taille des clauses. Il faut exhiber une valeur qui va décroitre strictement. On ne peut pas prendre le nombre de littéraux car de fait la procédure introduit à chaque étape une nouvelle variable et deux littéraux supplémentaires. On ne peut pas non plus dire que la taille de la plus grande clause diminue car il peut y en avoir plusieurs de taille maximale auquel cas la taille maximale ne change pas durant l'itération. L'entité la plus simple est de considérer le nombre total de littéraux dans les clauses de 4 littéraux ou plus. On remplace une clause de $n+2$ littéraux par une clause de 3 littéraux et une clause de $n+1$ littéraux donc si on ne considère que les clauses de 4 littéraux ou plus, le nombre total de littéraux a diminué strictement de 1.*

2. Soit $l_1 \vee l_2 \vee l_3 \vee C$ une clause avec l_1, l_2, l_3 des littéraux et C une clause non vide. Soit p une variable qui n'apparaît pas dans $l_1 \vee l_2 \vee l_3 \vee C$ montrer que

   (a) $l_1 \vee l_2 \vee p, \neg p \vee l_3 \vee C \models l_1 \vee l_2 \vee l_3 \vee C$

   (b) si I est une interprétation qui rend vraie $I \models l_1 \vee l_2 \vee l_3 \vee C$ alors il est possible de trouver une valeur booléenne b pour la variable p telle que $I, p \mapsto b \models (l_1 \vee l_2 \vee p) \wedge (\neg p \vee l_3 \vee C)$

   Correction :

   (a) Pour le premier cas, on peut remarquer que la règle de résolution entre les deux clauses $l_1 \vee l_2 \vee p$ et $\neg p \vee l_3 \vee C$ sur la variable propositionnelle p donne justement la clause $l_1 \vee l_2 \vee l_3 \vee C$ et donc la correction de la règle de résolution nous donne le résultat souhaité. On peut aussi refaire le raisonnement si une interprétation I est telle que $I \models l_1 \vee l_2 \vee p$ et $I \models \neg p \vee l_3 \vee C$ on a : soit $I \models p$, et alors $I \not\models \neg p$ et on en déduit $I \models l_3 \vee C$, soit $I \not\models p$, et alors $I \models l_1 \vee l_2$ dans les deux cas $I \models l_1 \vee l_2 \vee l_3 \vee C$

   (b) Supposons que $I \models l_1 \vee l_2 \vee l_3 \vee C$, on a alors que soit $I \models l_1 \vee l_2$ soit $I \models l_3 \vee C$.

— *si $I \models l_1 \vee l_2$, on prend $b = F$ et on a $I, p \mapsto b \models (l_1 \vee l_2 \vee p) \wedge (\neg p \vee l_3 \vee C)$*
— *sinon $I \models l_3 \vee C$, on prend $b = V$ et on a $I, p \mapsto b \models (l_1 \vee l_2 \vee p) \wedge (\neg p \vee l_3 \vee C)$*
dans les deux cas on a bien trouvé une interprétation qui étend I pour la nouvelle variable p qui rend vraies les deux clauses générées.

3. On note $\mathcal{E}'$ l'ensemble des clauses $\mathcal{C}$ obtenues à la fin de la procédure :
   (a) montrer que $\mathcal{E}' \models \mathcal{E}$ et si I est une interprétation telle que $I \models \mathcal{E}$ alors il est possible de trouver une interprétation I' des nouvelles variables introduites par la procédure telle que $I \cup I' \models \mathcal{E}'$.
   Indication : on pourra raisonner par récurrence sur le nombre d'itérations de la procédure
   Correction : *On raisonne par récurrence sur le nombre d'itérations pour obtenir $\mathcal{E}'$ à partir de $\mathcal{E}$, on montre les deux directions en même temps $\mathcal{E}' \models \mathcal{E}$ et si I est une interprétation telle que $I \models \mathcal{E}$ alors il est possible de trouver une interprétation I' des nouvelles variables introduites par la procédure telle que $I \cup I' \models \mathcal{E}'$.*
   — *s'il n'y a aucune itération (cas où toutes les clauses de $\mathcal{E}$ ont moins de 3 littéraux) alors $\mathcal{E}' = \mathcal{E}$ et le résultat est trivial.*
   — *soit n quelconque, On suppose le résultat vrai pour tout système qui termine en n étapes.*
   Soit $\mathcal{E}$ qui termine en $n+1$ étapes alors $\mathcal{E}$ contient une clause $l_1 \vee l_2 \vee l_3 \vee C$ avec au moins 4 littéraux, choisie pour la première étape. On note $\mathcal{E}_1$ l'ensemble privé de cette clause. Le système $\mathcal{E}$ est transformé en un système $\mathcal{E}_1 \cup \{l_1 \vee l_2 \vee p, \neg p \vee l_3 \vee C\}$, avec p une variable qui n'apparaît pas dans $\mathcal{E}$.
   Par hypothèse de récurrence on a que $\mathcal{E}' \models \mathcal{E}_1 \cup \{l_1 \vee l_2 \vee p, \neg p \vee l_3 \vee C\}$ et tout modèle de $\mathcal{E}_1 \cup \{l_1 \vee l_2 \vee p, \neg p \vee l_3 \vee C\}$ s'étend en un modèle de $\mathcal{E}'$.
   Par la question précédente, on a que $l_1 \vee l_2 \vee p, \neg p \vee l_3 \vee C \models l_1 \vee l_2 \vee l_3 \vee C$ et donc par transitivité de la conséquence logique : $\mathcal{E}' \models \mathcal{E}_1 \cup \{l_1 \vee l_2 \vee l_3 \vee C\}$, et donc $\mathcal{E}' \models \mathcal{E}$.
   Réciproquement si $I \models \mathcal{E}$ alors $I \models \mathcal{E}_1$ et $I \models l_1 \vee l_2 \vee l_3 \vee C$. Par la question précédente il existe une interprétation de p telle que $I, p \mapsto b \models (l_1 \vee l_2 \vee p) \wedge (\neg p \vee l_3 \vee C)$. Comme p n'apparaît pas dans $\mathcal{E}$, on a que $I, p \mapsto b \models \mathcal{E}_1$ et donc par hypothèse de récurrence, il existe une interprétation des nouvelles variables I'_1 telle que $I, p \mapsto b \cup I'_1 \models \mathcal{E}'$ et donc en prenant $I' = I'_1, p \mapsto b$, on obtient le résultat attendu.
   (b) Si $\mathcal{E}'$ est insatisfiable, que peut-on en déduire pour $\mathcal{E}$?
   Correction : *Les ensembles de clauses $\mathcal{E}'$ et $\mathcal{E}$ sont equi-satisfiables. En effet, si $\mathcal{E}'$ est satisfiable alors $\mathcal{E}$ l'est trivialement puisque $\mathcal{E}' \models \mathcal{E}$. Réciproquement, si $\mathcal{E}$ est satisfiable alors soit I une interprétation telle que $I \models \mathcal{E}$. Le résultat précédent nous dit qu'on peut étendre I en une interprétation qui rend vraie $\mathcal{E}'$, donc $\mathcal{E}'$ est satisfiable.*
   Par contraposée, on en déduit que si $\mathcal{E}'$ est insatisfiable alors il en est de même pour $\mathcal{E}$
4. Si on part d'une ensemble qui contient une seule clause de n littéraux avec $n > 3$, combien l'ensemble résultat contient-il de clauses ?
   Correction : *Chaque itération crée une nouvelle clause de taille 3 et une clause de taille $n-1$, on a donc à la fin $n-2$ clauses.*

Bonus Peut-on adapter cette méthode pour produire un ensemble de clauses equi-satisfiables avec moins de deux littéraux ?

Correction : *Non, la même méthode ne peut fonctionner, en effet une clause à 3 littéraux* $l_1 \vee l_2 \vee l_3$ *serait transformée en deux clauses* $l_1 \vee l_2 \vee p$ *et* $\neg p \vee l_3$*, mais on voit qu'on a toujours une clause à trois littéraux et donc le processus de transformation ne s'arrête pas. De fait la question de la satisfiabilité pour des clauses avec trois littéraux au plus reste, comme le cas général, un problème NP-complet alors que la satisfiabilité d'un ensemble de clauses de deux littéraux se ramène à un problème de calcul de composantes fortement connexes dans un graphe et se résoud en temps linéaire.*

Rappel des règles du système G (propositionnel)

hypothèse	$(\textsc{Hyp})\dfrac{}{A,\Gamma \vdash \Delta, A}$	
	gauche	droite
$\bot$	$\dfrac{}{\bot,\Gamma \vdash \Delta}$	$\dfrac{\Gamma \vdash \Delta}{\Gamma \vdash \Delta,\bot}$
$\top$	$\dfrac{\Gamma \vdash \Delta}{\top,\Gamma \vdash \Delta}$	$\dfrac{}{\Gamma \vdash \Delta,\top}$
$\neg$	$\dfrac{\Gamma \vdash \Delta, A}{\neg A,\Gamma \vdash \Delta}$	$\dfrac{A,\Gamma \vdash \Delta}{\Gamma \vdash \Delta,\neg A}$
$\wedge$	$\dfrac{A, B,\Gamma \vdash \Delta}{A \wedge B,\Gamma \vdash \Delta}$	$\dfrac{\Gamma \vdash \Delta, A \quad \Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, A \wedge B}$
$\vee$	$\dfrac{A,\Gamma \vdash \Delta \quad B,\Gamma \vdash \Delta}{A \vee B,\Gamma \vdash \Delta}$	$\dfrac{\Gamma \vdash \Delta, A, B}{\Gamma \vdash \Delta, A \vee B}$
$\Rightarrow$	$\dfrac{\Gamma \vdash \Delta, A \quad B,\Gamma \vdash \Delta}{A \Rightarrow B,\Gamma \vdash \Delta}$	$\dfrac{A,\Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, A \Rightarrow B}$