Chapitre 8

Logique propositionnelle

Introduction.

D'un point de vue formel, une logique est définie par une *syntaxe*, c'est-à-dire la donnée d'un ensemble de symboles et de règles. Il s'agit donc d'un langage (dont les mots sont appelés les formules logiques), à qui on associe une *sémantique* permettant d'attribuer une valeur (le vrai ou le faux, par exemple) aux symboles et aux formules.

Dans ce cours, nous nous intéresserons exclusivement à la logique propositionnelle bi-valuée. Une formule propositionnelle sera constituée uniquement de variables propositionnelles et de connecteurs logiques et ne pourra prendre que deux valeurs : VRAI ou FAUX.

Il existe aussi des logiques propositionnelles tri-valuées, par exemple la logique de Lukasiewicz avec trois états : VRAI, FAUX ou INDÉTERMINÉ ; ainsi qu'une logique des prédicats, complétant la logique propositionnelle par des quantificateurs, mais ces notions ne sont pas au programme.

Le calcul propositionnel bi-valué tient une grande place en informatique ; du fait que nos processeurs sont essentiellement constitués de portes binaires du type de celles que l'on va étudier dans ce chapitre.

1 Syntaxe des formules propositionnelles

1.1 Syntaxe

Définition.

Pour définir le langage de la logique des propositions, on utilise un alphabet Σ constitué :

- d'un ensemble dénombrable $\mathcal{V} = \{v_1, v_2, ..., v_n, ...\}$ dont les éléments seront appelés **variables propositionnelles**.
- d'un ensemble de **connecteurs logiques**, dont les principaux sont :
  - un connecteur unaire (d'arité 1) : $\neg$ appelé le connecteur logique de **négation** (qui se lit « non ») ;
  - quatre connecteurs binaires (d'arité 2), notés respectivement $\wedge$, $\vee$, $\implies$ et $\iff$ et appelés respectivement les connecteurs logiques de **conjonction** (« et »), **disjonction** (« ou »), **implication** et **équivalence**.
- et de parenthèses.

Remarque. Dans la suite, nous nous restreindrons au cas d'un ensemble de variables $\mathcal{V}$ fini.

Nous pouvons alors définir par induction la notion de formule propositionnelle (ou expression logique ou formule logique) :

Définition.

Soient n appartenant à $\mathbb{N}^*$ et $\mathcal{V} = \{v_1, ..., v_n\}$ un ensemble fini de variables propositionnelles.
Nous définissons l'ensemble $\mathcal{F}$ des **formules propositionnelles** sur $\mathcal{V}$ de manière inductive par :

- Cas de base : Toute variable est une formule.
- Étape d'induction :
  - Si P est une formule, alors $(\neg P)$ est une formule.
  - Si P_1 et P_2 sont des formules, alors $(P_1 \wedge P_2)$, $(P_1 \vee P_2)$, $(P_1 \implies P_2)$ et $(P_1 \iff P_2)$ sont aussi des formules.

Exemple. Si $\mathcal{V} = \{p, q, r, s\}$, alors $(((\neg p) \wedge q) \iff (r \vee p))$ est une formule propositionnelle sur $\mathcal{V}$.

Remarque. L'expression infixe des formules logiques nécessite bien évidemment l'usage de parenthèses pour lever toute ambiguïté sémantique, mais les règles de priorité usuelles entre opérateurs permettent d'éviter d'en écrire un trop grand nombre. Nous adopterons les règles d'écriture suivantes :

- suppression des parenthèses de début et de fin ;
- l'opérateur de négation $\neg$ a priorité sur les autres ;
- $\wedge$ et $\vee$ sont prioritaires sur $\implies$ et $\iff$.

Exemple. Prenons toujours $\mathcal{V} = \{p, q, r, s\}$. Alors $(\neg p \wedge q) \iff (r \vee p)$ est une formule propositionnelle sur $\mathcal{V}$.

1.2 Représentation arborescente

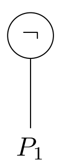
Définition.

La définition inductive amène à une **représentation arborescente** des formules propositionnelles, selon le principe suivant :

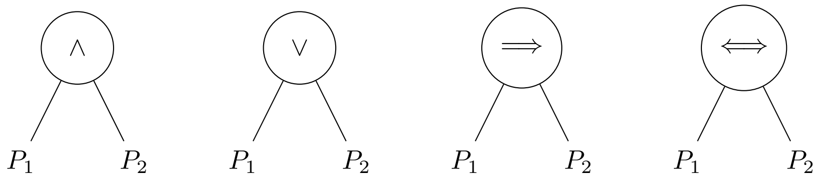
- Cas de base : L'arbre associé à une variable $p \in \mathcal{V}$ est une feuille

$$\boxed{p}$$

  contenant le nom de la variable.
- Étape d'induction : Pour toutes formules propositionnelles P_1 et P_2 sur $\mathcal{V}$,
  - L'arbre associé à $\neg P_1$ est un noeud $\neg$ ayant pour fils P_1 :



— Les arbres associés respectivement à $(P_1 \wedge P_2)$, $(P_1 \vee P_2)$, $(P_1 \implies P_2)$, et $(P_1 \iff P_2)$ sont des noeuds $\wedge$ ou $\vee$ ou $\implies$ ou $\iff$ ayant pour fils gauche l'arbre associé à P_1 et pour fils droit l'arbre associé à P_2 :



On peut définir la **hauteur** et la **taille** d'une formule logique comme la hauteur et la taille de l'arbre associé. On appelle **sous-formule** de P tout sous-arbre de P (P étant confondue avec l'arbre associé à P).

Exercice. On considère toujours $\mathcal{V} = \{p, q, r, s\}$.
Construire la représentation arborescente de $(\neg p \wedge q) \iff (r \vee p)$ puis donner sa hauteur et sa taille.

Propriété 1 (Arbre associé à une formule propositionnelle)

Le parcours en profondeur infixe de l'arbre associé à une formule propositionnelle reconstitue son écriture avec la syntaxe définie ci-avant.

Remarque. Il existe d'autres écritures d'une formule logique. Par exemple, l'écriture postfixe de Lukasiewick est celle obtenue par un parcours en profondeur postfixe de son arbre associé.
On rappelle que, d'après le chapitre 7, l'écriture de Lukasiewick détermine de manière unique l'arbre et donc la formule logique associée. Les parenthèses y sont donc inutiles, contrairement au parcours infixe.

Exercice. On considère toujours $\mathcal{V} = \{p, q, r, s\}$. Donner l'écriture de Lukasiewick de $(\neg p \wedge q) \iff (r \vee p)$.

2 Sémantique des formules propositionnelles

2.1 Valuations

Nous allons maintenant définir la sémantique d'une formule propositionnelle, c'est-à-dire le sens qu'on lui donne. La valeur de vérité d'une formule se définit comme l'interprétation de cette formule, une fois que l'on s'est fixé la valeur de vérité des variables propositionnelles.

Définition.

Soient n appartenant à $\mathbb{N}^*$, $\mathcal{V} = \{v_1, ..., v_n\}$ un ensemble fini de variables propositionnelles et $\mathcal{F}$ l'ensemble des formules propositionnelles sur $\mathcal{V}$.

Une **valuation** est une distribution de valeurs de vérité aux variables propositionnelles, c'est-à-dire une fonction de $\mathcal{V}$ vers $\mathbb{B} = \{0, 1\}$ où 0 représente FAUX, aussi noté F, et 1 représente VRAI, aussi noté V.

Exemple. La fonction $v : \{p \mapsto F; q \mapsto F; r \mapsto V; s \mapsto F\}$ est une valuation sur $\mathcal{V} = \{p, q, r, s\}$.

Propriété 2 (Nombre de valuations)

Il existe 2^n valuations différentes sur un ensemble de variables de cardinal n.

Définition.

Soient n appartenant à $\mathbb{N}^*$, $\mathcal{V} = \{v_1, ..., v_n\}$ un ensemble fini de variables propositionnelles, $\mathcal{F}$ l'ensemble des formules propositionnelles sur $\mathcal{V}$ et v une valuation définie sur $\mathcal{V}$.

Nous définissons la **fonction d'évaluation** sous v sur $\mathcal{F}$, notée E_v, de la manière inductive suivante :

- Cas de base : Pour toute variable propositionnelle $p \in \mathcal{V}$, $E_v(p) = v(p)$.
- Étape d'induction :
  - La négation $\neg$ s'interprète par la négation logique :
  $$E_v(P) = VRAI \text{ ssi } E_v(\neg P) = FAUX.$$
  - La conjonction $\wedge$ s'interprète comme le « et » logique :
  $$E_v(P_1 \wedge P_2) = VRAI \text{ ssi } (E_v(P_1) = VRAI \text{ et } E_v(P_2) = VRAI).$$
  - La disjonction $\vee$ s'interprète comme le « ou » logique :
  $$E_v(P_1 \vee P_2) = VRAI \text{ ssi } (E_v(P_1) = VRAI \text{ ou } E_v(P_2) = VRAI).$$
  - L'implication $\implies$ s'interprète comme l'implication logique :
  $$E_v(P_1 \implies P_2) = VRAI \text{ ssi } (E_v(P_1) = FAUX \text{ ou } E_v(P_2) = VRAI).$$
  - L'équivalence $\iff$ s'interprète comme l'équivalence logique :
  $$E_v(P_1 \iff P_2) = VRAI \text{ ssi } E_v(P_1) = E_v(P_2).$$

Exemple. Avec la valuation $\{p \mapsto F; q \mapsto F; r \mapsto V; s \mapsto F\}$ sur $\mathcal{V} = \{p, q, r, s\}$, la valeur de vérité de la formule $(\neg p \wedge q) \iff (r \vee p)$ est donnée par la dernière case du tableau suivant :

p	q	r	s	$\neg p$	$\neg p \wedge q$	$r \vee p$	$(\neg p \wedge q) \iff (r \vee p)$
F	F	V	F	V	F	V	F

2.2 Tables de vérité

À toute formule propositionnelle P sur $\mathcal{F}$, nous associons une table de vérité, c'est-à-dire un tableau à double entrée tel que :

- chaque ligne du tableau correspond à une valuation de $\mathcal{V}$ (soit 2^n lignes),
- les n premières colonnes correspondent aux différentes valeurs de vérité des variables propositionnelles $v_1, \ldots, v_n$ correspondant à la valuation v.
- la dernière colonne correspond à l'évaluation de P sous la valuation v décrite sur la ligne.

Il est possible d'ajouter des colonnes intermédiaires de calcul.

Exemples. Voici les tables de vérité des formules élémentaires sur deux variables v_1 et v_2 :

v_1	v_2	$\neg v_1$	$v_1 \wedge v_2$	$v_1 \vee v_2$	$v_1 \implies v_2$	$v_1 \iff v_2$
F	F	V	F	F	V	V
F	V	V	F	V	V	F
V	F	F	F	V	F	F
V	V	F	V	V	V	V

Par l'intermédiaire de leurs tables de vérité, il est possible de définir 16 connecteurs binaires différents. Aux quatre opérateurs déjà définis on peut rajouter le XOR (« ou exclusif ») souvent noté $\oplus$, le $NAND$ (« non et ») et le NOR (« non ou »), peu usités en mathématique mais beaucoup plus en informatique, et définis par l'intermédiaire des tables de vérité suivantes :

v_1	v_2	$v_1\ XOR\ v_2$	$v_1\ NAND\ v_2$	$v_1\ NOR\ v_2$
F	F	F	V	V
F	V	V	V	F
V	F	V	V	F
V	V	F	F	F

2.3 Satisfiabilité

Définition.

Soient n appartenant à $\mathbb{N}^*$, $\mathcal{V} = \{v_1, ..., v_n\}$ un ensemble fini de variables propositionnelles, $\mathcal{F}$ l'ensemble des formules propositionnelles sur $\mathcal{V}$ et P une formule propositionnelle sur $\mathcal{V}$.

- Nous dirons que la formule P est **satisfiable** si et seulement s'il existe au moins une valuation v de $\mathcal{V}$ telle que $E_v(P) = VRAI$, c'est-à-dire si la dernière colonne de la table de vérité de P contient au moins une valeur "VRAI".
- Un **modèle** pour la formule P est une valuation v de $\mathcal{V}$ qui rend la formule vraie, c'est-à-dire telle que $E_v(P) = VRAI$.

Exemple. Pour la formule P définie comme $(\neg p \wedge q) \iff (r \vee p)$, on remarque que :

- La valuation $\{p \mapsto F; q \mapsto F; r \mapsto V; s \mapsto F\}$ n'est pas un modèle.
- La valuation $\{p \mapsto F; q \mapsto V; r \mapsto V; s \mapsto F\}$ est un modèle.

Exercice. On pose $\mathcal{V} = \{x, y\}$. Donner l'ensemble des modèles pour la formule $\neg(x \wedge y) \implies \neg y$.

Remarque. La question de la satisfiabilité : « est-ce que tel problème possède au moins une solution ? » est fondamentale dans ce cours et en logique en général.

2.4 Tautologies, antilogies

Définition.

Soient n appartenant à $\mathbb{N}^*$, $\mathcal{V} = \{v_1, ..., v_n\}$ un ensemble fini de variables propositionnelles, $\mathcal{F}$ l'ensemble des formules propositionnelles sur $\mathcal{V}$ et P une formule propositionnelle sur $\mathcal{V}$.

- Nous dirons que la formule P est une **tautologie** (ou un théorème) si et seulement si pour toute valuation v de $\mathcal{V}$, $E_v(P) = VRAI$, c'est-à-dire si la dernière colonne de la table de vérité de P ne contient que des valeurs "VRAI".
- Nous dirons que la formule P est une **antilogie** (ou une contradiction) si et seulement si pour toute valuation v de $\mathcal{V}$, $E_v(P) = FAUX$, c'est-à-dire si la dernière colonne de la table de vérité de P ne contient que des valeurs "FAUX".

Remarque. Une antilogie est donc une formule propositionnelle qui n'est pas satisfiable.

Exemples.

- Les formules $a \vee \neg a$ (*le principe du tiers exclu*) et $(a \implies b) \wedge (b \implies c) \implies (a \implies c)$ (*la transitivité de l'implication*) sont des tautologies.
- La formule $a \wedge \neg a$ est une antilogie (*le principe de non contradiction*).
- La formule propositionnelle $P = (\neg p \wedge q) \iff (r \vee p)$ n'est ni une tautologie, ni une antilogie.

Remarque. Pour simplifier les écritures, nous allons dorénavant ajouter à la syntaxe (c'est-à-dire à l'alphabet Σ) une formule tautologique notée $\top$, et une formule antilogique notée $\bot$.

Exercice. Laquelle de ces deux formules est une tautologie ?

$$((a \implies b) \implies a) \implies a \qquad\qquad ((a \implies b) \implies a) \implies b$$

3 Équivalence sémantique et conséquence logique

3.1 Équivalence sémantique

Définition.

On dit que deux formules propositionnelles P_1 et P_2 sont **sémantiquement équivalentes**, et on note $P_1 \equiv P_2$, si pour toute valuation v,

$$E_v(P_1) = E_v(P_2).$$

Autrement dit, P_1 et P_2 sont **sémantiquement équivalentes** si leurs tables de vérité coïncident.

Attention.

Il ne faut pas pas confondre l'équivalence syntaxique avec l'équivalence sémantique !
Ainsi, $P_1 \iff P_2$ est une formule propositionnelle, alors que $P_1 \equiv P_2$ est une proposition. La valeur de cette dernière ne dépend pas du contexte.

Propriété 3 (de l'équivalence sémantique)

L'**équivalence sémantique** est une **relation d'équivalence** sur $\mathcal{F}$, c'est-à-dire qu'elle est :

- **Réflexive :** pour toute formule propositionnelle P,
$$P \equiv P.$$
- **Symétrique :** pour toutes formules propositionnelles P et Q,
$$P \equiv Q \iff Q \equiv P.$$
- **Transitive :** pour toutes formules propositionnelles P, Q et R,
$$\begin{cases} P \equiv Q \\ Q \equiv R \end{cases} \implies P \equiv R.$$

Propriété 4 (Caractérisation de l'équivalence sémantique)

Deux formules propositionnelles P_1 et P_2 sont **sémantiquement équivalentes** si et seulement si $(P_1 \iff P_2)$ est une **tautologie**.

Exercice. Vérifier en dressant les tables de vérités que :

1. $v_1 \implies v_2 \equiv \neg v_1 \vee v_2$

2. $v_1 \iff v_2 \equiv (v_1 \implies v_2) \wedge (v_2 \implies v_1)$

3. $v_1 \; NAND \; v_2 \equiv \neg(v_1 \wedge v_2)$

4. $v_1 \; NOR \; v_2 \equiv \neg(v_1 \vee v_2)$

5. $v_1 \oplus v_2 \equiv \neg(v_1 \iff v_2)$

3.2 Règles de calculs de l'équivalence sémantique

Lemme 5 (Compatibilité des connecteurs logiques avec l'équivalence sémantique)

Soient n appartenant à $\mathbb{N}^*$, $\mathcal{V} = \{v_1, ..., v_n\}$ un ensemble fini de variables propositionnelles, $\mathcal{F}$ l'ensemble des formules propositionnelles sur $\mathcal{V}$ et P_1, P_2, Q_1 et Q_2 des formules propositionnelles sur $\mathcal{V}$ telles que :

$$\begin{cases} P_1 \equiv P_2 \\ Q_1 \equiv Q_2 \end{cases}$$

Alors :

$$\begin{cases} \neg P_1 \equiv \neg P_2 \\ P_1 \wedge Q_1 \equiv P_2 \wedge Q_2 \\ P_1 \vee Q_1 \equiv P_2 \vee Q_2 \\ P_1 \Longrightarrow Q_1 \equiv P_2 \Longrightarrow Q_2 \\ P_1 \iff Q_1 \equiv P_2 \iff Q_2 \end{cases}$$

Théorème 6 (Invariance par substitution)

Soient n appartenant à $\mathbb{N}^*$, $\mathcal{V} = \{v_1, ..., v_n\}$ un ensemble fini de variables propositionnelles et $\mathcal{F}$ l'ensemble des formules propositionnelles sur $\mathcal{V}$.

Soient P_1 et P_2 deux formules propositionnelles sur $\mathcal{V}$ ne faisant intervenir que les variables propositionnelles $q_1, q_2, \ldots, q_k \in \mathcal{V}$ et $Q_1, Q_2, \ldots, Q_k$ des formules propositionnelles sur $\mathcal{V}$.

Notons :

- P_1' la formule propositionnelle obtenue à partir de P_1 en remplaçant q_1 par Q_1, q_2 par Q_2, ..., q_k par Q_k ;
- P_2' la formule propositionnelle obtenue à partir de P_2 en remplaçant q_1 par Q_1, q_2 par Q_2, ..., q_k par Q_k.

Si P_1 et P_2 sont sémantiquement équivalentes, alors P_1' et P_2' sont sémantiquement équivalentes.

3.3 Règles de la déduction naturelle

Propriété 7 (Connecteurs $\wedge$ et $\vee$)

Soient n appartenant à $\mathbb{N}^*$, $\mathcal{V} = \{v_1, ..., v_n\}$ un ensemble fini de variables propositionnelles, $\mathcal{F}$ l'ensemble des formules propositionnelles sur $\mathcal{V}$ et P_1, P_2 et P_3 des formules propositionnelles sur $\mathcal{V}$.

- **Élement neutre**

$$P_1 \wedge \top \equiv P_1 \qquad P_1 \vee \bot \equiv P_1$$

- **Élement absorbant**

$$P_1 \wedge \bot \equiv \bot \qquad P_1 \vee \top \equiv \top$$

- **Commutativité**

$$P_1 \wedge P_2 \equiv P_2 \wedge P_1 \qquad P_1 \vee P_2 \equiv P_2 \vee P_1$$

- **Associativité**

$$(P_1 \wedge P_2) \wedge P_3 \equiv P_1 \wedge (P_2 \wedge P_3) \qquad (P_1 \vee P_2) \vee P_3 \equiv P_1 \vee (P_2 \vee P_3)$$

- **Idempotence**

$$P_1 \wedge P_1 \equiv P_1 \qquad P_1 \vee P_1 \equiv P_1$$

Propriété 8 (Relations entre $\wedge$ et $\vee$)

Soient n appartenant à $\mathbb{N}^*$, $\mathcal{V} = \{v_1, ..., v_n\}$ un ensemble fini de variables propositionnelles, $\mathcal{F}$ l'ensemble des formules propositionnelles sur $\mathcal{V}$ et P_1, P_2 et P_3 des formules propositionnelles sur $\mathcal{V}$.

- **Subsomption**
$$P_1 \vee (P_1 \wedge P_2) \equiv P_1 \qquad P_1 \wedge (P_1 \vee P_2) \equiv P_1$$
- **Distributivité**
$$P_1 \vee (P_2 \wedge P_3) \equiv (P_1 \vee P_2) \wedge (P_1 \vee P_3) \qquad P_1 \wedge (P_2 \vee P_3) \equiv (P_1 \wedge P_2) \vee (P_1 \wedge P_3)$$

Propriété 9 (Négation)

Soient n appartenant à $\mathbb{N}^*$, $\mathcal{V} = \{v_1, ..., v_n\}$ un ensemble fini de variables propositionnelles, $\mathcal{F}$ l'ensemble des formules propositionnelles sur $\mathcal{V}$ et P_1 et P_2 des formules propositionnelles sur $\mathcal{V}$.

- **Négation de la négation**
$$\neg(\neg P_1) \equiv P_1$$
- **Principe du tiers exclu**
$$P_1 \vee \neg P_1 \equiv \top$$
- **Principe de non contradiction**
$$P_1 \wedge \neg P_1 \equiv \bot$$
- **Lois de De Morgan**
$$\neg(P_1 \wedge P_2) \equiv \neg P_1 \vee \neg P_2 \qquad \neg(P_1 \vee P_2) \equiv \neg P_1 \wedge \neg P_2$$

Propriété 10 (Raisonnement mathématique)

Soient n appartenant à $\mathbb{N}^*$, $\mathcal{V} = \{v_1, ..., v_n\}$ un ensemble fini de variables propositionnelles, $\mathcal{F}$ l'ensemble des formules propositionnelles sur $\mathcal{V}$ et P_1 et P_2 des formules propositionnelles sur $\mathcal{V}$.

- **Décomposition de l'implication**
$$P_1 \implies P_2 \equiv \neg P_1 \vee P_2$$
- **Symétrie de l'équivalence syntaxique**
$$P_1 \iff P_2 \equiv P_2 \iff P_1$$
- **Disjonction de cas**
$$(P_1 \implies P_2) \wedge (\neg P_1 \implies P_2) \equiv P_2$$
- **Contraposition**
$$P_1 \implies P_2 \equiv \neg P_2 \implies \neg P_1$$
- **Raisonnement par l'absurde**
$$\neg P_1 \implies \bot \equiv P_1$$

3.4 Système complet de connecteurs

Définition.

Soient n appartenant à $\mathbb{N}^*$, $\mathcal{V} = \{v_1, ..., v_n\}$ un ensemble fini de variables propositionnelles et $\mathcal{F}$ l'ensemble des formules propositionnelles sur $\mathcal{V}$.

Un système $\mathcal{S}$ de connecteurs logiques est dit **complet** si et seulement si toute formule propositionnelle P sur $\mathcal{V}$ est sémantiquement équivalente à une formule propositionnelle sur $\mathcal{V}$ ne faisant intervenir que les connecteurs de $\mathcal{S}$.

Exercice. Montrer que $\mathcal{S} = \{\neg, \wedge, \vee\}$ est un système complet de connecteurs.

3.5 Conséquence logique

Définition.

Soient n appartenant à $\mathbb{N}^*$, $\mathcal{V} = \{v_1, ..., v_n\}$ un ensemble fini de variables propositionnelles et $\mathcal{F}$ l'ensemble des formules propositionnelles sur $\mathcal{V}$.

- On dit qu'une formule propositionnelle P_2 est une **conséquence logique** d'une formule P_1, et on note $P_1 \models P_2$, si tout modèle de P_1 est un modèle de P_2.
- On dit qu'une formule propositionnelle Q est une **conséquence logique** d'un ensemble de formules $\{P_1, P_2, \dots, P_n\}$, et on note $P_1, P_2, \dots, P_n \models Q$, si toute valuation qui est un modèle commun à $P_1, P_2, \dots, P_n$ est aussi un modèle de Q.

Exemples. On a par exemple :
- $\perp \models P_1$
- $P_1 \models \top$
- $P_1 \wedge P_2 \models P_1$
- $P_1 \models P_1 \vee P_2$

Attention.

Il ne faut pas confondre l'implication syntaxique avec la conséquence logique !
Ainsi, $P_1 \implies P_2$ est une formule propositionnelle, alors que $P_1 \models P_2$ est une proposition. La valeur de cette dernière ne dépend pas du contexte.

Propriété 11 (Conséquence logique)

Soient n appartenant à $\mathbb{N}^*$, $\mathcal{V} = \{v_1, ..., v_n\}$ un ensemble fini de variables propositionnelles et $\mathcal{F}$ l'ensemble des formules propositionnelles sur $\mathcal{V}$. Pour toutes formules $P_1, P_2, \dots, P_n$ et Q, on a les propriétés suivantes :

- $P_1, P_2, ..., P_n \models Q$ si et seulement si la formule $P_1 \wedge P_2 \wedge ... \wedge P_n \implies Q$ est une tautologie.
- $P_1, P_2, ..., P_n \models Q$ si et seulement si la formule $P_1 \wedge P_2 \wedge ... \wedge P_n \wedge \neg Q$ est insatisfiable (antilogie).

Exercice. Soient P_1, P_2 et P_3 des formules propositionnelles. Prouver les propriétés suivantes :

1. $P_1 \wedge P_2 \models P_1$ (élimination du « et »).

2. $P_1 \models P_1 \vee P_2$ (introduction du « ou »).

3. $P_1 \wedge (P_1 \implies P_2) \models P_2$ (élimination de l'implication).

4. $(P_1 \implies P_2) \wedge (P_2 \implies P_3) \models (P_1 \implies P_3)$ (transitivité de l'implication).

4 Formes normales d'une formule propositionnelle

4.1 Formes normales

On considère toujours dans cette partie n appartenant à $\mathbb{N}^*$, $\mathcal{V} = \{v_1, ..., v_n\}$ un ensemble fini de variables propositionnelles et $\mathcal{F}$ l'ensemble des formules propositionnelles sur $\mathcal{V}$.

La relation d'équivalence sémantique sur $\mathcal{F}$ nous permet de partitionner $\mathcal{F}$ en classes d'équivalence. Nous allons nous entendre pour choisir une formule propositionnelle représentant la classe d'équivalence à laquelle elle appartient.

Ainsi, pour savoir si deux formules propositionnelles sont sémantiquement équivalentes, autrement dit pour savoir si deux formules propositionnelles sont dans la même classe d'équivalence, nous déterminerons le représentant de la classe d'équivalence contenant chaque formule et nous n'aurons plus qu'à regarder si les deux représentants sont syntaxiquement égaux.

Définition.

On appelle **littéral** toute formule propositionnelle de la forme p ou $\neg p$, p appartenant à $\mathcal{V}$.

Définition.

- On appelle **disjonction de formules propositionnelles** toute formule propositionnelle de la forme $P_1 \vee P_2 \vee \ldots \vee P_n$, où $P_1, P_2, \ldots, P_n$ sont des formules propositionnelles sur $\mathcal{V}$.
- On appelle **conjonction de formules propositionnelles**, toute formule propositionnelle de la forme $P_1 \wedge P_2 \wedge \ldots \wedge P_n$, où $P_1, P_2, \ldots, P_n$ sont des formules propositionnelles sur $\mathcal{V}$.

Lemme 12 (Extension des lois de De Morgan)

Soient n appartenant à $\mathbb{N}^*$ et $P_1, P_2, \ldots, P_n$ des formules propositionnelles sur $\mathcal{V}$. Alors :

$$\neg\big(P_1 \vee P_2 \vee ... \vee P_n\big) \equiv \neg P_1 \wedge \neg P_2 \wedge ... \wedge \neg P_n\,,$$

$$\neg\big(P_1 \wedge P_2 \wedge ... \wedge P_n\big) \equiv \neg P_1 \vee \neg P_2 \vee ... \vee \neg P_n\,.$$

On retiendra que la négation d'une disjonction est une conjonction, et réciproquement.

Lemme 13 (Extension de la distributivité)

Soient p et q appartenant à $\mathbb{N}^*$, $P_1, P_2, \ldots, P_p$ et $Q_1, Q_2, \ldots, Q_q$ des formules propositionnelles sur $\mathcal{V}$. Alors :

$$\Big(\bigwedge_{i=1}^{p} P_i\Big) \vee \Big(\bigwedge_{j=1}^{q} Q_j\Big) \equiv \bigwedge_{\substack{1\le i\le p\\ 1\le j\le q}} (P_i \vee Q_j)\,,$$

$$\Big(\bigvee_{i=1}^{p} P_i\Big) \wedge \Big(\bigvee_{j=1}^{q} Q_j\Big) \equiv \bigvee_{\substack{1\le i\le p\\ 1\le j\le q}} (P_i \wedge Q_j)\,.$$

Théorème 14 (Théorème fondamental)

(1) Toute formule propositionnelle sur $\mathcal{V}$ est sémantiquement équivalente à une conjonction de disjonctions de littéraux :

$$\bigwedge_i \Big(\bigvee_j \ell_{i,j}\Big).$$

Une conjonction de disjonctions de littéraux sera appelée une **forme normale conjonctive**.

(2) Toute formule propositionnelle sur $\mathcal{V}$ est sémantiquement équivalente à une disjonction de conjonctions de littéraux :

$$\bigvee_i \Big(\bigwedge_j \ell_{i,j}\Big).$$

Une disjonction de conjonctions de littéraux sera appelée une **forme normale disjonctive**.

Preuve.

$\square$

Exercice. On considère une formule propositionnelle P sur $\mathcal{V} = \{p, q, r\}$ dont la table de vérité est :

p	q	r	P
F	F	F	V
F	F	V	V
F	V	F	F
F	V	V	F
V	F	F	F
V	F	V	F
V	V	F	V
V	V	V	V

.

Déterminer une forme normale disjonctive sémantiquement équivalente à P.

Méthode.

Afin de déterminer une forme normale conjonctive sémantiquement équivalente à P :

1. On détermine une forme normale disjonctive sémantiquement équivalente à $\neg P$.
2. Par négation, on détermine une conjonction de disjonction de littéraux sémantiquement équivalente à P.

Exercice. On considère toujours la formule propositionnelle P donnée dans l'exercice précédent.
Déterminer une forme normale disjonctive sémantiquement équivalente à P.

4.2 Formes normales canoniques

Définition.

- On appelle **minterme** sur $\mathcal{V}$ toute conjonction de littéraux dans laquelle chaque variable propositionnelle de $\mathcal{V}$ intervient une et une seule fois.
- On appelle **maxterme** sur $\mathcal{V}$ toute disjonction de littéraux dans laquelle chaque variable propositionnelle de $\mathcal{V}$ intervient une et une seule fois.

Exemple. On pose $\mathcal{V} = \{p, q, r\}$. Alors :

- $p \wedge \neg q \wedge \neg r$ est un minterme sur $\mathcal{V}$ ("min" dans le sens "le moins de chance d'être évalué à VRAI").
- $p \vee \neg q \vee r$ est un maxterme sur $\mathcal{V}$ ("max" dans le sens "le plus de chance d'être évalué à VRAI").

Définition.

Soit P une formule propositionnelle sur $\mathcal{V}$ ne faisant intervenir que les variables propositionnelles $q_1, q_2, \ldots, q_k$.

- On appelle **minterme relatif à** P tout minterme sur $\{q_1, q_2, \ldots, q_k\}$.
- On appelle **maxterme relatif à** P tout maxterme sur $\{q_1, q_2, \ldots, q_k\}$.

Théorème 15 (Formes normales canoniques)

Soit P une formule propositionnelle sur $\mathcal{V}$. Alors :

(1) P est sémantiquement équivalente à une disjonction de mintermes relatifs à P, unique à l'ordre des facteurs près, appelée **la forme normale disjonctive canonique de** P.

(2) P est sémantiquement équivalente à une conjonction de maxtermes relatifs à P, unique à l'ordre des facteurs près, appelée **la forme normale conjonctive canonique de** P.

Méthode.

1. La forme normale disjonctive canonique d'une formule P se lit très simplement sur sa table de vérité en considérant les lignes satisfaisant la formule (comme on l'a vu dans l'exercice précédent). On montre alors que la formule est équivalente à la disjonction des formules caractérisant chaque ligne.
   Ceci est à la fois un mode pratique pour l'obtention de forme normale disjonctive canonique et une démonstration du théorème précédent.
2. Pour obtenir la forme normale conjonctive canonique de P, il suffit d'appliquer la méthode précédente à $\neg P$ puis les lois de De Morgan.

Exercice. Soit $\mathcal{V} = \{v_1, v_2\}$ un ensemble fini de deux variables propositionnelles et $\mathcal{F}$ l'ensemble des formules propositionnelles sur $\mathcal{V}$. Donner les formes normales conjonctives et disjonctives canoniques de $\iff$ et de $\oplus$.

Méthode.
Pour obtenir la forme normale disjonctive canonique d'une formule, on peut également appliquer l'algorithme suivant :

1. On exprime la formule uniquement avec $\neg$, $\wedge$ et $\vee$.
2. On "descend" le connecteur $\neg$ à l'aide des lois de De Morgan.
3. On utilise la distributivité de $\wedge$ par rapport à $\vee$, autrement dit on "descend" les connecteurs $\wedge$ par rapport aux connecteurs $\vee$.
4. On utilise le tiers exclu pour faire apparaître les variables absentes dans les clauses.
5. On utilise à nouveau la distributivité de $\wedge$ par rapport à $\vee$.

Le mot "descendre" correspond à la descente dans la représentation arborescente des formules logiques.

Exercice. Soit $\mathcal{V} = \{v_1, v_2, v_3\}$ un ensemble fini de trois variables propositionnelles et $\mathcal{F}$ l'ensemble des formules propositionnelles sur $\mathcal{V}$. Donner les formes normales conjonctives et disjonctives canoniques de

$$P = \neg(v_1 \wedge \neg(v_2 \wedge v_3))$$

1. En utilisant les lois de De Morgan et autres règles naturelles de déduction.

2. En utilisant une table de vérité.

Propriété 16 (Caractérisation de l'équivalence sémantique)

Deux formules propositionnelles sont sémantiquement équivalentes si et seulement si leurs formes normales canoniques sont syntaxiquement égales.

5 Problème SAT

Le problème `SAT` est le problème de décision qui consiste à savoir si une formule logique est satisfiable :

Définition.

- Soit $k \in \mathbb{N}^*$. Une formule propositionnelle est dite en $k-$FNC si elle est en forme normale conjonctive et que chaque clause disjonctive comporte au plus k littéraux.
- On donne un nom aux problèmes de décision suivants :
  — `SAT` : Soit φ une formule propositionnelle, φ est-elle satisfiable ?
  — $k-$`SAT` : Soit φ une formule propositionnelle en $k-$FNC, φ est-elle satisfiable ?

L'algorithme naïf pour vérifier qu'une formule est satisfiable consiste à tester toutes les distributions de vérités ce qui est donc exponentiel (mauvais !) par rapport au nombre n des variables (car il y a 2^n valuations).

La question fondamentale consiste à savoir si ce problème de décision peut être résolu en temps polynomial. On cherche donc à savoir s'il appartient à l'une des deux classes importantes de problèmes de décision suivantes :

- La classe `P` constituée des problèmes de décision qui peuvent être résolus en temps polynomial par rapport à la taille de l'entrée.
- La classe `NP` constituée des problèmes de décision pour lesquels on peut vérifier la validité d'une solution en temps polynomial.

Il est évident que nous disposons de l'inclusion $\mathtt{P} \subset \mathtt{NP}$. Le problème de l'inclusion réciproque est un des plus grands défis de l'informatique théorique et à ce titre figure dans la liste des problèmes du millénaire.

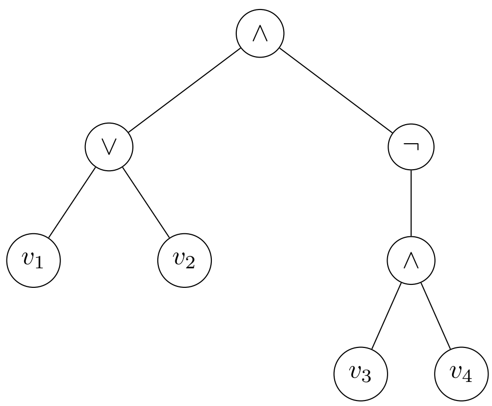
Parmi les problèmes de la classe `NP`, on distingue certains d'entre eux, qualifiés de problèmes `NP`-complets. D'une certaine façon, ce sont les plus difficiles des problèmes `NP` : si on connaissait une solution polynomiale pour l'un d'eux, on en connaîtrait une pour tout problème de la classe, et on aurait l'égalité $\mathtt{P} = \mathtt{NP}$. Le problème `SAT` a la particularité d'avoir été le premier problème `NP`-complet trouvé. Ce résultat est maintenant connu sous le nom de théorème de Cook.

☞ Voir le sujet X-ENS 2016 sur la satisfiabilité des formules booléennes.

6 Exercices

Exercice 1

Donner une écriture parenthésée de la formule de représentation arborescente



Exercice 2

1. Donner l'écriture arborescente et l'écriture de Lukasiewicz des formules suivantes :

   (a) $v_1 \wedge (\neg v_2 \vee v_3)$ (b) $\neg(v_1 \wedge (v_2 \vee \neg v_3))$

2. Donner l'écriture arborescente de la formule d'écriture de Lukasiewicz $v_1 \neg v_2 \wedge v_3 v_4 v_5 \vee \vee \wedge$

Exercice 3

Soit v_1, v_2, v_3 et v_4 quatre variables. Déterminer si les formules suivantes sont des tautologies ou des antilogies :

(a) $(v_1 \implies v_2) \vee (v_3 \implies v_4)$ (b) $(v_1 \implies v_2) \vee (v_2 \implies v_3)$

Exercice 4

Soit v_1, v_2 et v_3 trois variables. Déterminer l'ensemble des modèles de :

$$(v_1 \wedge v_2) \iff (v_1 \vee v_3)$$

Exercice 5

Montrer que les connecteurs $NAND$ et NOR ne sont pas associatifs, puis que le connecteur $\oplus$ est associatif.

Exercice 6

Prouver, en utilisant les tables de vérité :

1. Les lois de De Morgan,
2. Le principe du tiers exclu,
3. La disjonction de cas et la contraposition,
4. Le raisonnement par l'absurde.

Exercice 7

Soient a, b et c trois variables propositionnelles.

1. Sans utiliser de table de vérité, simplifier la formule $P_1 = \neg(a \wedge b) \wedge (a \vee \neg b) \wedge (a \vee b)$
2. Démontrer que les propositions suivantes sont des tautologies, sans utiliser de table de vérité :

$$P_2 = (a \wedge b) \vee c \vee (\neg b \wedge \neg c) \vee (\neg a \wedge \neg c), \qquad P_3 = a \vee (\neg b \wedge \neg c) \vee (\neg a \wedge c) \vee (b \wedge \neg c)$$

Exercice 8

On interroge un logicien qui dit toujours la vérité sur sa vie sentimentale. À la question : « est-il vrai que si vous aimez Marie, alors vous aimez Anne ? » il répond :
– si c'est vrai, alors j'aime Marie ;
– si j'aime Marie, alors c'est vrai.
Qu'en concluez-vous ?

Exercice 9

Une nouvelle série de composants informatiques dédiés au raisonnement logique a été conçue de manière à faciliter la détection de pannes. Chaque processeur effectue des raisonnements logiques et peut être, soit en état de fonctionnement normal, soit en état de panne. Il se comporte alors de la manière suivante :

- un processeur en état de fonctionnement normal ne peut affirmer que des propositions vraies ;
- un processeur en état de panne ne peut affirmer que des propositions fausses.

Un ordinateur est composé de trois processeurs qui possèdent la même mémoire, donc les mêmes connaissances. Périodiquement, un ingénieur vient interroger l'ordinateur pour déterminer si certains processeurs sont en état de panne.

Lors d'une séance de test, l'ingénieur pose les deux questions suivantes au processeur n^o 1 :

- « Est-ce que les processeurs n^o 2 et n^o 3 sont en état de fonctionnement normal ? »
- « Est-ce que le processeur n^o 2 est en état de fonctionnement normal ? »

Le processeur n^o 1 répond à la première question : « Les processeurs n^o 2 et n^o 3 sont en état de fonctionnement normal. »

Puis, répond à la deuxième question : « Le processeur n^o 2 est en état de panne. »

Nous noterons p_1 (respectivement p_2 et p_3) la variable propositionnelle « le processeur n^o 1 (respectivement n^o 2 et n^o 3) est en état de panne ».

Nous supposerons que l''état des trois processeurs ne peut pas changer entre les réponses aux deux questions.

1. (a) Exprimer la réponse R_2 à la deuxième question sous la forme d'une formule propositionnelle (en envisageant la possibilité que le processeur n^o 1 soit en panne ou non) et donner sa taille, sa hauteur, ainsi que son écriture postfixe.
   (b) Exprimer la réponse R_1 à la première question sous la forme d'une formule propositionnelle et donner sa taille, sa hauteur, ainsi que son écriture postfixe.
2. (a) Donner la table de vérité des formules R_1 et R_2, puis de $R_1 \wedge R_2$.
   (b) En déduire l'état de chaque processeur.
3. (a) Prouver, par équivalence sémantique (c'est-à-dire en utilisant les règles de la déduction naturelle, et non une table de vérité), que
$$R_1 \wedge R_2 \equiv p_1 \wedge \neg p_2 \wedge p_3$$
   (b) Que peut-on en conclure ?

Exercice 10

Lors de ses aventures au pays des merveilles rapportées par Lewis Carroll, Alice est souvent accompagnée par le chat de Cheshire. Ce félin énigmatique s'exprime sous la forme d'affirmations logiques qui sont toujours vraies.

Alice se trouve dans un corridor dont toutes les portes à sa taille sont fermées. La seule porte ouverte est nettement trop petite pour qu'elle puisse l'emprunter. Une étagère est fixée au-dessus de cette porte. Le chat dit à Alice : "L'un des flacons posés sur cette étagère contient un liquide qui te permettra de prendre une taille plus adéquate. Mais attention, les autres flacons peuvent contenir un poison fatal".

Trois flacons sont effectivement posés sur l'étagère. Le premier est rouge, le second jaune, le troisième bleu. Une étiquette est collée sur chaque flacon. Alice lit l'inscription figurant sur chaque étiquette :

- Flacon rouge : le flacon jaune contient un poison, le flacon bleu ne contient pas un poison.
- Flacon jaune : si le flacon rouge contient un poison, alors le flacon bleu aussi.
- Flacon bleu : je ne contiens pas un poison, mais au moins l'un des deux autres flacons contient un poison.

Nous noterons R, J et B les variables propositionnelles correspondant au fait que les flacons rouge, jaune et bleu contiennent un poison. Nous noterons I_R, I_J et I_B les propositions correspondant aux inscriptions sur les flacons.

1. Exprimer I_R, I_J et I_B sous la forme de formules du calcul des propositions dépendant de R, J et B .
2. Traduire les questions suivantes en formules du calcul des propositions puis résoudre les problèmes posés en utilisant le calcul des propositions (formules de De Morgan et/ou tables de vérité).
   (a) Les inscriptions sur les trois flacons sont-elles compatibles ?
   (b) Est-ce que les inscriptions sont dépendantes, c'est-à-dire est-ce qu'une inscription est une conséquence logique de la conjonction des deux autres ? Si oui, préciser la (ou les) dépendance(s).
   (c) Dans le cas où aucun des trois flacons ne contient un poison, est-ce qu'une ou plusieurs inscriptions sont fausses ? Si oui, laquelle ou lesquelles ?
   (d) Si les trois inscriptions sont vraies, est-ce qu'un ou plusieurs flacons contiennent un poison ? Si oui, lequel ou lesquelles ?
   (e) Si les flacons ne contenant pas un poison ont une inscription vraie et les flacons contenant un poison ont une inscription fausse, est-ce qu'un ou plusieurs flacons ne contiennent pas un poison ? Si oui, lequel ou lesquels ?

Exercice 11

De nombreux travaux sont réalisés en Intelligence Artificielle pour construire un programme qui imite le raisonnement humain et qui soit capable de réussir le test de Turing, c'est-à-dire qu'il ne puisse pas être distingué d'un être humain dans une conversation à l'aveugle. Vous êtes chargé(e)s de vérifier la correction des réponses données par un tel programme lors des tests de bon fonctionnement. Dans le scénario de test considéré, le comportement attendu est le respect de la règle suivante : pour chaque question, le programme répondra par trois affirmations dont une seule sera correcte.

Nous noterons A_1, A_2, A_3 les propositions associées aux affirmations effectuées par le programme.

1. Représenter le comportement attendu sous la forme d'une formule du calcul des propositions qui dépend de A_1, A_2 et A_3.

Premier cas :
Vous demandez au programme : Quels éléments doivent contenir les aliments que je dois consommer pour préserver ma santé ?
Il répond les affirmations suivantes :
A_1 : Consommez au moins des aliments qui contiennent des glucides, mais pas de lipides !
A_2 : Si vous consommez des aliments qui contiennent des glucides, alors ne consommez pas d'aliments qui contiennent des lipides !
A_3 : Ne consommez aucun aliment qui contient des lipides !

Nous noterons g, respectivement l, les variables propositionnelles qui correspondent au fait de consommer des aliments qui contiennent des glucides, respectivement des lipides.

2. Exprimer A_1, A_2 et A_3 sous la forme de formules du calcul des propositions. Ces formules peuvent dépendre des variables g et l.
3. En utilisant le calcul des propositions (résolution avec les formules de De Morgan), déterminer ce que doivent contenir les aliments que vous devrez consommer pour préserver votre santé.

Second cas :
Vous demandez au programme : Quelles activités dois-je pratiquer si je veux préserver ma santé ?
Suite à une coupure de courant, la dernière affirmation est interrompue.
A_1 : Ne faites des activités sportives que si vous prenez également du repos !
A_2 : Si vous ne faites pas d'activité intellectuelle, alors ne prenez pas de repos !
A_3 : Prenez du repos ou faites des activités ... !

Nous noterons s, i et r les variables propositionnelles qui correspondent au fait de faire des activités sportives, des activités intellectuelles et de prendre du repos.

4. Exprimer A_1, A_2 et A_3 sous la forme de formule du calcul des propositions. Ces formules peuvent dépendre de s, i et r.
5. En utilisant le calcul des propositions (résolution avec les tables de vérité), déterminer quelle(s) activité(s) vous devez pratiquer pour préserver votre santé.

Exercice 12

Dans une association de jeunes détectives, les membres s'entraînent à résoudre des problèmes logiques. Ceux-ci respectent les règles suivantes : « Lors d'une conversation, un même membre aura un comportement constant : il dira toujours la vérité, ou ne dira jamais la vérité. » et « Une conversation ne doit pas être absurde. »

1. Soient n membres intervenant dans une même conversation. Chaque membre est représenté par une variable propositionnelle m_i avec $i \in [\![1, n]\!]$ qui représente le fait que ce membre dit, ou ne dit pas la vérité. Chaque membre fait une seule déclaration représentée par la formule propositionnelle D_i.
   Représenter le respect des règles dans cette conversation sous la forme d'une formule du calcul des propositions dépendant des variables m_i et D_i avec $i \in [\![1, n]\!]$.
2. Vous assistez à deux conversations sur la véracité des déclarations de deux groupes de trois membres de cette association. Nous nommerons A, B et C les participants de la première conversation.
   - A : « Les seuls qui disent la vérité ici sont C et moi. »
   - B : « C ne dit pas la vérité. »
   - C : « Soit B dit la vérité. Soit A ne dit pas la vérité. »

   Nous noterons a, b et c les variables propositionnelles associées au fait que A, B et C disent respectivement la vérité.
   Nous noterons D_A, D_B et D_C les formules du calcul des propositions associées respectivement aux déclarations de A, B et C dans la conversation.
   (a) Représenter les déclarations de la première conversation sous la forme de formules du calcul des propositions D_A, D_B et D_C dépendant des variables a, b et c.
   (b) En utilisant le calcul des propositions (résolution avec les tables de vérité), débusquez au moins un menteur !
3. Nous nommerons D, E et F les participants de la seconde conversation.
   - D : « Personne ne doit croire F. »
   - E : « D et F disent toujours la vérité. »
   - F : « E dit la vérité. »

Nous noterons d, e et f les variables propositionnelles associées au fait que D, E et F disent respectivement la vérité.
Nous noterons D_D, D_E et D_F les formules du calcul des propositions associées respectivement aux déclarations de D, E et F dans la seconde conversation.

(a) Représenter les déclarations de la seconde conversation sous la forme de formules du calcul des propositions D_D, D_E et D_F dépendant des variables d, e et f

(b) En utilisant le calcul des propositions (résolution avec les formules de De Morgan), déterminer si D, E et F disent, ou ne disent pas, la vérité.

Exercice 13

On désire étendre la logique propositionnelle bi-valuée classique (notée dans la suite $\mathcal{L}_2$) de sorte à prendre en compte, en plus de V (vrai) et F (faux), une troisième valeur de vérité, notée I (pour indéterminée).

La **logique de Lukasiewicz**, notée $\mathcal{L}_3$, est une telle logique tri-valuée dans laquelle le connecteur de négation ($\neg$), le connecteur de conjonction ($\wedge$), le connecteur de disjonction ($\vee$) et le connecteur d'implication ($\Rightarrow$) sont définis par les tables de vérité suivantes :

x	$\neg x$
V	F
F	V
I	I

$\neg x$

$x \backslash y$	V	F	I
V	V	F	I
F	F	F	F
I	I	F	I

$x \wedge y$

$x \backslash y$	V	F	I
V	V	V	V
F	V	F	I
I	V	I	I

$x \vee y$

$x \backslash y$	V	F	I
V	V	F	I
F	V	V	V
I	V	I	V

$x \Rightarrow y$

On dit que deux propositions sont **équivalentes** dans une logique donnée $\mathcal{L}$ si elles ont même table de vérité dans cette logique.

1. Indiquer une proposition P_1 équivalente dans $\mathcal{L}_2$ à la proposition « $x \vee y$ » et n'utilisant pas d'autres connecteurs que $\neg$ et $\wedge$. Les propositions P_1 et « $x \vee y$ » sont-elles équivalentes dans $\mathcal{L}_3$?
2. Indiquer une proposition P_2 équivalente dans $\mathcal{L}_2$ à la proposition « $x \Rightarrow y$ » et n'utilisant pas d'autres connecteurs que $\neg$ et $\wedge$. Les propositions P_2 et « $x \Rightarrow y$ » sont-elles équivalentes dans $\mathcal{L}_3$?
3. Établir (en détaillant son obtention) la table de vérité dans $\mathcal{L}_3$ de la proposition P_3 suivante :

$$((x \Rightarrow y) \wedge ((\neg x) \Rightarrow y)) \Rightarrow y.$$

   Cette proposition est-elle une tautologie dans $\mathcal{L}_3$ (la définition d'une tautologie dans la logique $\mathcal{L}_3$ reste la même que dans la logique $\mathcal{L}_2$) ?
4. On appelle **littéral** une variable propositionnelle ou sa négation. Dans $\mathcal{L}_2$, toute proposition logique admet une forme normale disjonctive (c'est-à-dire une disjonction de conjonctions de littéraux) qui lui est équivalente.
   (a) Établir une proposition P_4 sous forme normale disjonctive équivalente dans $\mathcal{L}_2$ à la proposition « $(x \vee y) \Rightarrow z$ ».
   (b) Les propositions P_4 et « $(x \vee y) \Rightarrow z$ » sont-elles équivalentes dans $\mathcal{L}_3$?
5. Peut-on faire un raisonnement par contraposition dans $\mathcal{L}_3$ (on justifiera la réponse) ?
