Exercice 1. Théorème de complétude

Question 1 – On fixe la formule logique A et on montre le résultat par récurrence sur $n \in \mathbb{N}$.

Initialisation. Pour $n = 0$, on remarque que :

$$V_0 = \varnothing \qquad |D_0| = 1 \qquad \forall \mu \in D_n : \Gamma_\mu = \varnothing.$$

Soit μ le seul élément de D_0. Si on suppose $\Gamma_\mu \vdash A$, on a directement $\varnothing \vdash A$, c'est à dire $\vdash A$.

Hérédité. On suppose la propriété vraie au rang $n \in \mathbb{N}$ et on la montre au rang $n+1$. Supposons $\Gamma_\mu \vdash A$ pour tout $\mu \in D_{n+1}$. Par l'hypothèse de récurrence, il nous suffit de montrer $\Gamma_\mu \vdash A$ pour tout $\mu \in D_n$.

Soit $\mu \in D_n$. On note $\mu_0 \in D_{n+1}$ et $\mu_1 \in D_{n+1}$ les distributions de vérités définies par :

$$\begin{cases} \mu_0(v_{n+1}) = 0 \\ \forall i \in [\![1, n]\!] : \mu_0(v_i) = \mu(v_i) \end{cases} \qquad \begin{cases} \mu_1(v_{n+1}) = 1 \\ \forall i \in [\![1, n]\!] : \mu_1(v_i) = \mu(v_i) \end{cases}$$

Par hypothèse, il existe un arbre de preuve $\mathcal{A}_0$ pour le séquent $\Gamma_{\mu_0} \vdash A$ et un arbre de preuve $\mathcal{A}_1$ pour le séquent $\Gamma_{\mu_1} \vdash A$. De plus, on a :

$$\Gamma_{\mu_0} = \Gamma_\mu \cup \{\neg v_{n+1}\} \qquad \Gamma_{\mu_1} = \Gamma_\mu \cup \{v_{n+1}\}$$

Construisons un arbre de preuve pour le séquent $\Gamma_\mu \vdash A$:

$$\dfrac{\dfrac{\mathcal{A}_1}{\Gamma_\mu, v_{n+1} \vdash A} \qquad \dfrac{\mathcal{A}_0}{\Gamma_\mu, \neg v_{n+1} \vdash A}}{\Gamma_\mu \vdash A}\ \text{te}$$

Ainsi, pour tout $\mu \in D_n$, le séquent $\Gamma_\mu \vdash A$ est prouvable. Par hypothèse de récurrence, le séquent $\vdash A$ est donc prouvable.

Question 2 – On remarque que :

- Si $|F| = 1$, alors $F = \bot$ ou $F = v$ avec v une variable propositionnelle.
- Si $|F| > 1$, alors $F = \neg F_1$ ou $F = F_1 \wedge F_2$ ou $F = F_1 \vee F_2$ ou $F = F_1 \to F_2$ avec F_1 et F_2 des formules logiques telles que $|F_1| < |F|$ et $|F_2| < |F|$.

Soit $n \in \mathbb{N}$ et $k \in \mathbb{N}^*$. Supposons la propriété vraie pour toute formule F' telle que $|F'| < k$ et montrons la pour une formule F telle que $|F| = k$.

$\star$ Si $F = \bot$, alors pour tout $\mu \in D_n$, on a $E_\mu(F) = 0$ et donc $\Gamma_\mu \vdash \neg F$ est bien prouvable :

$$\dfrac{\dfrac{}{\Gamma_\mu, F \vdash \bot}\ \text{ax}}{\Gamma_\mu \vdash \neg F}\ \neg_i$$

$\star$ On suppose que $F = v$ avec v une variable propositionnelle. Pour tout $\mu \in D_n$, par définition de Γ_μ :

$$\begin{cases} \text{Si } E_\mu(v) = 0, \text{ alors } \neg v \in \Gamma_\mu, \text{ donc } \Gamma_\mu \vdash \neg F \text{ est prouvable en appliquant la règle de l'axiome.} \\ \text{Si } E_\mu(v) = 1, \text{ alors } v \in \Gamma_\mu, \text{ donc } \Gamma_\mu \vdash F \text{ est prouvable en appliquant la règle de l'axiome.} \end{cases}$$

$\star$ On suppose que $F = \neg F_1$ et que $E_\mu(F) = 0$ (resp. $E_\mu(F) = 1$). Alors, par définition de E_μ, on a $E_\mu(F_1) = 1$ (resp. $E_\mu(F_1) = 0$). Par l'hypothèse de récurrence, il existe un arbre de preuve $\mathcal{A}$ pour $\Gamma_\mu \vdash F_1$ (resp. $\Gamma_\mu \vdash \neg F_1$). Construisons un arbre de preuve pour $\Gamma_\mu \vdash \neg\neg F_1$ (resp. $\Gamma_\mu \vdash \neg F_1$) :

$$\dfrac{\dfrac{\dfrac{\dfrac{\mathcal{A}}{\Gamma_\mu \vdash F_1}}{\Gamma_\mu, \neg F_1 \vdash F_1}\ \text{aff} \qquad \dfrac{}{\Gamma_\mu, \neg F_1 \vdash \neg F_1}\ \text{ax}}{\Gamma_\mu, \neg F_1 \vdash \bot}\ \neg_e}{\Gamma_\mu \vdash \neg\neg F_1}\ \neg_i \qquad\qquad \dfrac{\mathcal{A}}{\Gamma_\mu \vdash \neg F_1}$$

⋆ On suppose que $F = F_1 \wedge F_2$ et que $E_\mu(F) = 0$. Alors, par définition de E_μ, on a $E_\mu(F_1) = 0$ ou $E_\mu(F_2) = 0$. On traite le cas où $E_\mu(F_1) = 0$ (l'autre cas est symétrique). Par l'hypothèse de récurrence, il existe un arbre de preuve $\mathcal{A}$ pour $\Gamma_\mu \vdash \neg F_1$. Construisons un arbre de preuve pour $\Gamma_\mu \vdash \neg(F_1 \wedge F_2)$:

$$
\begin{prooftree}
\AxiomC{}
\RightLabel{ax}
\UnaryInfC{$\Gamma_\mu, F_1 \wedge F_2 \vdash F_1 \wedge F_2$}
\RightLabel{$\wedge_e^g$}
\UnaryInfC{$\Gamma_\mu, F_1 \wedge F_2 \vdash F_1$}
\AxiomC{$\mathcal{A}$}
\UnaryInfC{$\Gamma_\mu \vdash \neg F_1$}
\RightLabel{aff}
\UnaryInfC{$\Gamma_\mu, F_1 \wedge F_2 \vdash \neg F_1$}
\RightLabel{$\neg_e$}
\BinaryInfC{$\Gamma_\mu, F_1 \wedge F_2 \vdash \bot$}
\RightLabel{$\neg_i$}
\UnaryInfC{$\Gamma_\mu \vdash \neg(F_1 \wedge F_2)$}
\end{prooftree}
$$

⋆ On suppose que $F = F_1 \wedge F_2$ et que $E_\mu(F) = 1$. Alors, par définition de E_μ, on a $E_\mu(F_1) = 1$ et $E_\mu(F_2) = 1$. Par l'hypothèse de récurrence, il existe un arbre de preuve $\mathcal{A}_1$ pour $\Gamma_\mu \vdash F_1$ et un arbre de preuve $\mathcal{A}_2$ pour $\Gamma_\mu \vdash F_2$. Construisons un arbre de preuve pour $\Gamma_\mu \vdash F_1 \wedge F_2$:

$$
\begin{prooftree}
\AxiomC{$\mathcal{A}_1$}
\UnaryInfC{$\Gamma_\mu \vdash F_1$}
\AxiomC{$\mathcal{A}_1$}
\UnaryInfC{$\Gamma_\mu \vdash F_2$}
\RightLabel{$\wedge_i$}
\BinaryInfC{$\Gamma_\mu \vdash F_1 \wedge F_2$}
\end{prooftree}
$$

⋆ On suppose que $F = F_1 \vee F_2$ et que $E_\mu(F) = 0$. Alors, par définition de E_μ, on a $E_\mu(F_1) = 0$ et $E_\mu(F_2) = 0$. Par l'hypothèse de récurrence, il existe un arbre de preuve $\mathcal{A}_1$ pour $\Gamma_\mu \vdash \neg F_1$ et un arbre de preuve $\mathcal{A}_2$ pour $\Gamma_\mu \vdash \neg F_2$. Construisons un arbre de preuve pour $\Gamma_\mu \vdash \neg(F_1 \vee F_2)$:

$$
\begin{prooftree}
\AxiomC{\boxed{\text{Arbre similaire}}}
\UnaryInfC{$\Gamma_\mu, F_1 \vee F_2, F_2 \vdash \bot$}
\AxiomC{}
\RightLabel{ax}
\UnaryInfC{$\Gamma_\mu, F_1 \vee F_2, F_1 \vdash F_1$}
\AxiomC{$\mathcal{A}_1$}
\UnaryInfC{$\Gamma_\mu \vdash \neg F_1$}
\RightLabel{aff}
\UnaryInfC{$\Gamma_\mu, F_1 \vee F_2 \vdash \neg F_1$}
\RightLabel{aff}
\UnaryInfC{$\Gamma_\mu, F_1 \vee F_2, F_1 \vdash \neg F_1$}
\RightLabel{$\neg_e$}
\BinaryInfC{$\Gamma_\mu, F_1 \vee F_2, F_1 \vdash \bot$}
\AxiomC{}
\RightLabel{ax}
\UnaryInfC{$\Gamma_\mu, F_1 \vee F_2 \vdash F_1 \vee F_2$}
\RightLabel{$\vee_e$}
\TrinaryInfC{$\Gamma_\mu, F_1 \vee F_2 \vdash \bot$}
\RightLabel{$\neg_i$}
\UnaryInfC{$\Gamma_\mu \vdash \neg(F_1 \vee F_2)$}
\end{prooftree}
$$

⋆ On suppose que $F = F_1 \vee F_2$ et que $E_\mu(F) = 1$. Alors, par définition de E_μ, on a $E_\mu(F_1) = 1$ ou $E_\mu(F_2) = 1$. On traite le cas où $E_\mu(F_1) = 1$ (l'autre cas est symétrique). Par l'hypothèse de récurrence, il existe un arbre de preuve $\mathcal{A}$ pour $\Gamma_\mu \vdash F_1$. Construisons un arbre de preuve pour $\Gamma_\mu \vdash F_1 \vee F_2$:

$$
\begin{prooftree}
\AxiomC{$\mathcal{A}$}
\UnaryInfC{$\Gamma_\mu \vdash F_1$}
\RightLabel{$\vee_i^g$}
\UnaryInfC{$\Gamma_\mu \vdash F_1 \vee F_2$}
\end{prooftree}
$$

⋆ On suppose que $F = F_1 \to F_2$ et que $E_\mu(F) = 0$. Alors, par définition de E_μ, on a $E_\mu(F_1) = 1$ et $E_\mu(F_2) = 0$. Par l'hypothèse de récurrence, il existe un arbre de preuve $\mathcal{A}_1$ pour $\Gamma_\mu \vdash F_1$ et un arbre de preuve $\mathcal{A}_2$ pour $\Gamma_\mu \vdash \neg F_2$. Construisons un arbre de preuve pour $\Gamma_\mu \vdash \neg(F_1 \to F_2)$:

$$
\begin{prooftree}
\AxiomC{$\mathcal{A}_1$}
\UnaryInfC{$\Gamma_\mu \vdash F_1$}
\RightLabel{aff}
\UnaryInfC{$\Gamma_\mu, F_1 \to F_2 \vdash F_1$}
\AxiomC{}
\RightLabel{ax}
\UnaryInfC{$\Gamma_\mu, F_1 \to F_2 \vdash F_1 \to F_2$}
\RightLabel{$\to_e$}
\BinaryInfC{$\Gamma_\mu, F_1 \to F_2 \vdash F_2$}
\AxiomC{$\mathcal{A}_2$}
\UnaryInfC{$\Gamma_\mu \vdash \neg F_2$}
\RightLabel{aff}
\UnaryInfC{$\Gamma_\mu, F_1 \to F_2 \vdash \neg F_2$}
\RightLabel{$\neg_e$}
\BinaryInfC{$\Gamma_\mu, F_1 \to F_2 \vdash \bot$}
\RightLabel{$\neg_i$}
\UnaryInfC{$\Gamma_\mu \vdash \neg(F_1 \to F_2)$}
\end{prooftree}
$$

⋆ On suppose que $F = F_1 \to F_2$ et que $E_\mu(F) = 1$. Alors, par définition de E_μ, on a $E_\mu(F_1) = 0$ ou $E_\mu(F_2) = 1$. Par l'hypothèse de récurrence, il existe un arbre de preuve $\mathcal{A}_1$ pour $\Gamma_\mu \vdash \neg F_1$ ou bien un arbre de preuve $\mathcal{A}_2$ pour $\Gamma_\mu \vdash F_2$. Dans les deux cas, construisons un arbre de preuve pour $\Gamma_\mu \vdash F_1 \to F_2$:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\mathcal{A}_1}{\Gamma_\mu \vdash \neg F_1}}{\Gamma_\mu, F_1 \vdash \neg F_1}\ \text{aff}}{\Gamma_\mu, F_1, \neg F_2 \vdash \neg F_1}\ \text{aff} \qquad \dfrac{}{\Gamma_\mu, F_1, \neg F_2 \vdash F_1}\ \text{ax}}{\Gamma_\mu, F_1, \neg F_2 \vdash \bot}\ \neg_e}{\Gamma_\mu, F_1 \vdash F_2}\ \bot_c}{\Gamma_\mu \vdash F_1 \rightarrow F_2}\ \rightarrow_i \qquad\qquad \dfrac{\dfrac{\dfrac{\mathcal{A}_2}{\Gamma_\mu, \vdash F_2}}{\Gamma_\mu, F_1 \vdash F_2}\ \text{aff}}{\Gamma_\mu \vdash F_1 \rightarrow F_2}\ \rightarrow_i$$

Question 3 – Soit F une tautologie et $n \in \mathbb{N}$ un entier tel que toutes les variables propositionnelles de F appartiennent à V_n. Pour tout $\mu \in D_n$, si on note $\mu' : V \rightarrow \{0,1\}$ une distribution de vérité dont μ est la restriction à V_n, alors $E_\mu(F) = E_{\mu'}(F) = 1$. D'après la question 2 : $\Gamma_\mu \vdash F$. D'après la question 1, $\vdash F$ est prouvable.

Question 4 – On le montre par récurrence sur k.

<u>Initialisation.</u> Pour $k = 1$, on suppose disposer d'un arbre de preuve $\mathcal{A}$ pour $\Gamma \vdash A_1$ et d'un arbre de preuve $\mathcal{B}$ pour $\Gamma, A_1 \vdash A$. Montrons $\Gamma \vdash A$:

$$\dfrac{\dfrac{\mathcal{A}}{\Gamma \vdash A_1} \qquad \dfrac{\dfrac{\mathcal{B}}{\Gamma, A_1 \vdash A}}{\Gamma \vdash A_1 \rightarrow A}\ \rightarrow_i}{\Gamma \vdash A}\ \rightarrow_e$$

<u>Hérédité.</u> Soit $k \geqslant 1$ tel que $\vee_{e,k}$ est admissible. Montrons que $\vee_{e,k+1}$ est admissible. On suppose disposer d'un arbre de preuve $\mathcal{A}$ pour $\Gamma \vdash \mathit{Disj}(A_1, \ldots, A_{k+1})$ et de $k+1$ arbres de preuves $\mathcal{B}_i$ pour $\Gamma, A_i \vdash A$ où $i \in [\![1, k+1]\!]$. Montrons que $\Gamma \vdash A$ est prouvable :

$$\dfrac{\dfrac{\mathcal{A}}{\Gamma \vdash \mathit{Disj}(A_1, \ldots, A_k) \vee A_{k+1}} \qquad \dfrac{\mathcal{C}}{\Gamma_1 \vdash A} \qquad \dfrac{\mathcal{B}_{k+1}}{\Gamma, A_{k+1} \vdash A}}{\Gamma \vdash A}\ \vee_e$$

où $\Gamma_1 = \Gamma \cup \{\mathit{Disj}(A_1, \ldots, A_k)\}$ et $\mathcal{C}$ l'arbre :

$$\dfrac{\dfrac{}{\Gamma_1 \vdash \mathit{Disj}(A_1, \ldots, A_k)}\ \text{ax} \qquad \dfrac{\dfrac{\mathcal{B}_1}{\Gamma, A_1 \vdash A}}{\Gamma_1, A_1 \vdash A}\ \text{aff} \qquad \ldots \qquad \dfrac{\dfrac{\mathcal{B}_k}{\Gamma, A_k \vdash A}}{\Gamma_1, A_k \vdash A}\ \text{aff}}{\Gamma_1 \vdash A}\ \vee_{e,k}$$

Question 5 – Supposons F valide dans T et posons $T' = T \cup \{\neg F\}$. Comme F est valide dans T, la théorie T' n'admet pas de modèle. Par le théorème de compacité, il existe $\Gamma' \subset T'$ fini tel que Γ' n'admet pas de modèle. Notons $A_1, \ldots, A_k$ les éléments de Γ'.

Comme Γ' n'admet pas de modèle, pour toute distribution de vérité μ, il existe $i \in [\![1, k]\!]$ tel que $E_\mu(A_i) = 0$. Ainsi, par définition de E_μ :

$$E_\mu\left(\mathit{Disj}(\neg A_1, \ldots, \neg A_k)\right) = 1$$

En d'autres termes la formule $\mathit{Disj}(\neg A_1, \ldots, \neg A_k)$ est une tautologie. Par la question 3, il existe un arbre de preuve $\mathcal{A}$ pour $\vdash \mathit{Disj}(\neg A_1, \ldots, \neg A_k)$.

Soit $\Gamma = \Gamma' \setminus \{\neg F\}$ (si $\neg F \notin \Gamma'$ alors $\Gamma = \Gamma'$). Construisons maintenant un arbre de preuve pour $\Gamma \vdash F$ ce qui conclura la démonstration :

$$\dfrac{\dfrac{\dfrac{\mathcal{A}}{\vdash \mathit{Disj}(\neg A_1, \ldots, \neg A_k)}}{\Gamma \vdash \mathit{Disj}(\neg A_1, \ldots, \neg A_k)}\ \text{aff} \qquad \dfrac{\mathcal{A}_1}{\Gamma, \neg A_1 \vdash F} \qquad \ldots \qquad \dfrac{\mathcal{A}_k}{\Gamma, \neg A_k \vdash F}}{\Gamma \vdash F}\ \vee_{e,k}$$

Il reste à construire les arbres $\mathcal{A}_i$. Si $A_i = \neg F$, alors l'arbre suivant convient :

$$\dfrac{\dfrac{\dfrac{}{\Gamma_1 \vdash \neg F}\,\text{ax} \qquad \dfrac{}{\Gamma_1 \vdash \neg\neg F}\,\text{ax}}{\Gamma_1 \vdash \bot}\,\neg_e}{\Gamma, \neg\neg F \vdash F}\,\bot_c \qquad \text{avec } \Gamma_1 = \Gamma \cup \{\neg\neg F, \neg F\}.$$

Sinon $A_i \neq \neg F$, donc $A_i \in \Gamma$ et l'arbre suivant convient :

$$\dfrac{\dfrac{\dfrac{}{\Gamma_2 \vdash A_i}\,\text{ax} \qquad \dfrac{}{\Gamma_2 \vdash \neg A_i}\,\text{ax}}{\Gamma_2 \vdash \bot}\,\neg_e}{\Gamma, \neg A_i \vdash F}\,\bot_c \qquad \text{avec } \Gamma_2 = \Gamma \cup \{\neg A_i, \neg F\}.$$

Question 6 – On le montre par double implication.

($\Rightarrow$) Supposons $T \vdash \bot$. Le théorème de correction sitpule que $\bot$ est valide dans T, c'est à dire que pour tout $\mu \in D$:

$$\text{Si } \mu \text{ est un modèle de } T, \text{ alors } E_\mu(\bot) = 1.$$

Or $E_\mu(\bot) = 0$ par définition de E_μ. Ainsi, T n'admet pas de modèle.

($\Leftarrow$) On montre la contraposée. Si $T \nvdash \bot$, alors d'après le théorème de complétude, $\bot$ n'est pas valide dans T. Par définition de “valide” :

$$\text{Il existe un modèle } \mu \text{ de } T \text{ tel que } E_\mu(\bot) = 0$$

En particulier, T admet un modèle.

Question 7 – Montrons que $T = V$ convient. Pour cela on utilise les question 2 et 6 :

→ Soit μ la distribution de vérité telle que $\mu(v) = 1$ pour tout V. On remarque que μ est un modèle de T, donc $T \nvdash \bot$ par la question 6.

→ Soit F une formule logique et $n \in \mathbb{N}$ un entier tel que toutes les variables propositionnelles de F sont dans V_n. Soit μ la distribution de vérité telle que $\mu(v) = 1$ pour tout $v \in V_n$, alors $E_\mu(F) \in \{0, 1\}$. Par la question 2, on a $\Gamma_\mu \vdash F$ ou $\Gamma_\mu \vdash \neg F$. Puisque $\Gamma_\mu \subset T$, on obtient $T \vdash F$ ou $T \vdash \neg F$.

Exercice 2. Théorème de compacité

Question 1 – On suppose par l'absurde que $T \cup \{v\}$ et $T \cup \{\neg v\}$ ne sont pas FS. Par définition, il existe $\Gamma_1 \subset T \cup \{v\}$ et $\Gamma_2 \subset T \cup \{\neg v\}$ des ensembles finis n'admettant pas de modèle. Comme T est FS, on a $v \in \Gamma_1$ et $\neg v \in \Gamma_2$. On pose :

$$\Gamma_1' = \Gamma_1 \setminus \{v\} \subset T \qquad \text{et} \qquad \Gamma_2' = \Gamma_2 \setminus \{\neg v\} \subset T$$

Soit μ un modèle de $\Gamma_1' \cup \Gamma_2'$. Si $\mu(v) = 0$, alors μ est un modèle de $\Gamma_1' \cup \Gamma_2' \cup \{\neg v\}$ et donc un modèle de Γ_2. Sinon, $\mu(v) = 1$ et donc μ est un modèle de Γ_1. Dans les deux cas, on a une contradiction avec l'hypothèse de départ.

Question 2 – On définit une suite de théories $(T_k)_{k\in\mathbb{N}}$ par récurrence sur k. On pose $T_0 = T$ et pour tout $k \geqslant 0$:

$$T_{k+1} = \begin{cases} T_k \cup \{v_k\} \text{ si } T_k \cup \{v_k\} \text{ est FS} \\ T_k \cup \{\neg v_k\} \text{ sinon} \end{cases}$$

D'après la question précédente, T_k est FS pour tout k. On pose :

$$T' = \bigcup_{k\in\mathbb{N}} T_k.$$

Il est clair que $T \subset T'$ et que pour toute variable $v \in V$, on a $v \in T'$ ou $\neg v \in T'$. Il reste à montrer que T' est FS. Soit $\Gamma \subset T'$ fini. Comme Γ est fini et que $T_0 \subset T_1 \subset T_2 \subset \ldots$, il existe $k_0 \in \mathbb{N}$ tel que $\Gamma \subset T_{k_0}$. Puisque T_{k_0} est FS, il existe un modèle de Γ.

Question 3 – On définit T' la théorie de la question précédente. Remarquons que si $v \in T'$, alors $\neg v \notin T'$. En effet, dans le cas contraire, on aurait $\{v, \neg v\} \subset T'$, ce qui contredit le fait que T' est FS en prenant $\Gamma = \{v, \neg v\}$.

Soit μ la distribution de vérité définie par :

$$\forall v \in V : \mu(v) = \begin{cases} 1 \text{ si } v \in T' \\ 0 \text{ si } \neg v \in T' \end{cases}$$

Montrons que μ est un modèle de T. Soit $F \in T$ et $k \in \mathbb{N}$ tel que toutes les variables de F appartiennent à $\{v_1, \ldots, v_k\}$. Pour tout $i \in [\![1, k]\!]$, on pose :

$$\ell_i = \begin{cases} v_i \text{ si } v_i \in T' \\ \neg v_i \text{ si } \neg v_i \in T' \end{cases}$$

Alors $\Gamma = \{F, \ell_1, \ell_2, \ldots, \ell_k\} \subset T'$ est fini et admet donc un modèle μ'. À cause de la présence des ℓ_i, μ et μ' coïncident sur $v_1, \ldots, v_k$. Ainsi :

$$E_\mu(F) = E_{\mu'}(F) = 1$$

Exercice 3. Admissibilité de la règle d'affaiblissement

Supposons qu'il existe un arbre de preuve $\mathcal{A}$ pour le séquent $\Gamma \vdash A$ et construisons un arbre de preuve pour $\Gamma, B \vdash A$. On le montre par récurrence sur la hauteur de $\mathcal{A}$.

Initialisation. Si $\mathcal{A}$ est de hauteur 0, alors $\mathcal{A}$ est de la forme :

$$\dfrac{}{\Gamma \vdash A}\ \text{ax}$$

Ce qui signifie que $A \in \Gamma$. Comme $A \in \Gamma \cup \{B\}$:

$$\dfrac{}{\Gamma, B \vdash A}\ \text{ax} \qquad \text{est un arbre de preuve pour } \Gamma, B \vdash A.$$

Hérédité. Sinon, $\mathcal{A}$ est de la forme :

$$\dfrac{\dfrac{\mathcal{B}_1}{\Gamma_1 \vdash A_1} \quad \dfrac{\mathcal{B}_2}{\Gamma_2 \vdash A_2} \quad \cdots \quad \dfrac{\mathcal{B}_n}{\Gamma_n \vdash A_n}}{\Gamma \vdash A}\ r$$

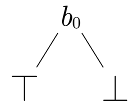
avec r la règle utilisée à la racine de $\mathcal{A}$ et $\mathcal{B}_1$, …, $\mathcal{B}_n$ des arbres de preuve. Pour tout $i \in [\![1, n]\!]$, l'hypothèse de récurrence assure l'existence d'un arbre de preuve $\mathcal{C}_i$ pour le séquent $\Gamma, B \vdash A_i$. Voici un arbre de preuve $\mathcal{A}'$ pour $\Gamma, B \vdash A$:

$$\dfrac{\dfrac{\mathcal{C}_1}{\Gamma_1, B \vdash A_1} \quad \dfrac{\mathcal{C}_2}{\Gamma_2, B \vdash A_2} \quad \cdots \quad \dfrac{\mathcal{C}_n}{\Gamma_n, B \vdash A_n}}{\Gamma, B \vdash A}\ r$$

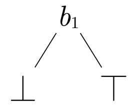
Pour écrire ce qui précède, on a vérifié exhaustivement pour les 11 règles de la logique classique (sauf l'affaiblissement et l'axiome) que si r est appliquée correctement à la racine de $\mathcal{A}$, alors r est appliquée correctement à la racine de $\mathcal{A}'$.

Exercice 4. Arbres de décision (concours X/ENS 1999)

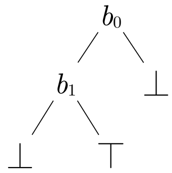
Question 1 – Pour π_0 :



Pour $\neg\pi_1$:



Pour $\pi_0 \wedge \neg\pi_1$, $\neg\pi_1 \wedge \pi_0$ et $\neg(\pi_0 \Rightarrow \pi_1)$, on obtient trois fois le même arbre :



Question 2 –

Idée. Dans un arbre de décision différent de $\top$, il y a au moins un $\bot$. En effet, il suffit de considérer un nœud interne de profondeur maximale, alors d'après la condition (C 2.1), l'un des fils est $\top$ et l'autre est $\bot$. On considère alors un chemin de la racine vers l'une des feuilles étiquetée par $\bot$. Lorsqu'on rencontre un b_i dans ce chemin, on pose $b_i = 1$ si le chemin va dans le sous-arbre gauche, et $b_i = 0$ sinon. D'après la condition (C 2.2), chaque b_i est rencontré au plus une fois. Les b_i qui ne se trouvent pas sur le chemin peuvent prendre une valeur arbitraire. Soit g la fonction représentée par l'arbre A. Lorsque les b_i prennent les valeurs décrites ci-dessus, on a $g(b_0, \ldots, b_{n-1}) = 0$ et donc g n'est pas constante égale à 1.

Preuve formelle. Montrons par récurrence forte décroissante finie sur $i \in [\![0, n-1]\!]$ que pour tout arbre A dont la racine est étiquetée par b_i, il existe $b_i, b_{i+1}, \ldots, b_{n-1} \in \mathcal{B}$ tels que pour tout $b_0, \ldots, b_{i-1} \in \mathcal{B}$:

$$g(b_0, \ldots, b_{n-1}) = 0$$

où $g : \mathcal{B}^n \to \mathcal{B}$ est la fonction booléenne représentée par A. Dans la suite, on note v et f les sous-arbres gauche et droit de A, ainsi que g, g' et g'' les fonctions représentées par A, v et f.

Soit $i \in [\![0, n-1]\!]$. On suppose la proposition vraie pour tout rang $j > i$ et on la montre au rang i.

- Si $v = \bot$, alors on pose $b_i = 1$ et $b_{i+1} = b_{i+2} = \ldots = b_{n-1} = 0$ (en fait ces booléens peuvent être quelconques). On a alors :

$$g(b_0, \ldots, b_{n-1}) = test(\pi_i, g', g'')(b_0, \ldots, b_{n-1}) = g'(b_0, \ldots, b_{n-1}) = 0$$

- Si $f = \bot$, on procède comme dans le cas précédente en posant $b_i = 0$.
- Si $v \neq \bot$ et $v \neq \top$, soit $j \in [\![i+1, n-1]\!]$ tel que la racine de v est étiquetée par b_j. D'après l'hypothèse de récurrence, il existe $b_j, \ldots, b_{n-1}$ tels que pour tout $b_0, \ldots, b_{j-1}$, on a $g'(b_0, \ldots, b_{n-1}) = 0$. On pose alors $b_i = 1$ et $b_{i+1} = \ldots = b_{j-1} = 0$ (en fait ces booléens peuvent être quelconques). Alors pour tout $b_0, \ldots, b_{i-1}$, on a bien $g(b_0, \ldots, b_{n-1}) = g'(b_0, \ldots, b_{n-1}) = 0$.
- Sinon $v = \top$ et $f \neq \bot$. D'après la condition (C 2.1), on a aussi $f \neq \top$. On procède comme dans le cas précédent en posant $b_i = 0$.

En conclusion, il n'existe pas d'arbre de hauteur $h \geqslant 1$ représentant la fonction constante égale à 1. L'arbre réduit à la feuille $\bot$ ne représente pas non plus la fonction constante égale à 1 d'où la proposition.

Question 3 – On remarque que l'égalité $g_{i\leftarrow 0} = g_{i\leftarrow 1}$ est équivalente à dire que la fonction g ne dépend pas de la variable b_i. En effet, si $g_{i\leftarrow 0} = g_{i\leftarrow 1}$ alors pour tout $b_0, \dots, b_{i-1}, b_{i+1}, \dots, b_{n-1}$:

$$\begin{aligned} g(b_0, \dots, b_{i-1}, 0, b_{i+1}, \dots, b_{n-1}) &= g_{i\leftarrow 0}(b_0, \dots, b_{i-1}, 0, b_{i+1}, \dots, b_{n-1}) \\ &= g_{i\leftarrow 1}(b_0, \dots, b_{i-1}, 0, b_{i+1}, \dots, b_{n-1}) \\ &= g(b_0, \dots, b_{i-1}, 1, b_{i+1}, \dots, b_{n-1}) \end{aligned}$$

En d'autres termes, pour tout $b_0, \dots, b_{n-1}$ et tout $x \in \mathcal{B}$:

$$g(b_0, \dots, b_{i-1}, b_i, b_{i+1}, \dots, b_{n-1}) = g(b_0, \dots, b_{i-1}, x, b_{i+1}, \dots, b_{n-1}).$$

Dans cette question, pour $i \in [\![0, n]\!]$ on note G_i l'ensemble des fonctions $g : \mathcal{B}^n \to \mathcal{B}$ qui ne dépendent d'aucune variable b_j avec $j < i$. Il nous suffit donc de montrer que toutes les fonctions de G_0 sont représentables par un arbre de décision. Montrons par récurrence décroissante sur i que toutes les fonctions de G_i sont représentables par un arbre de décision ne contenant pas les variables $b_0, \dots, b_{i-1}$.

Initialisation. Si $i = n$ alors une fonction $g \in G_n$ ne dépend d'aucune des variables b_i. En d'autres termes, la fonction g est constante. Si elle est constante égale à 1, elle est représentable par l'arbre $\top$, sinon elle est constante égale à 0 donc elle est représentable par l'arbre $\perp$. Dans les deux cas, g est représentable par un arbre de décision ne contenant pas les variables $b_0, \dots, b_{n-1}$.

Hérédité. Soit $i \in [\![0, n-1]\!]$. On suppose la propriété vraie au rang $i+1$ et on la montre au rang i. Pour tout $g \in G_i$:

- Si $g_{i\leftarrow 0} = g_{i\leftarrow 1}$ alors la fonction g ne dépend pas de la variable b_i et donc $g \in G_{i+1}$. D'après l'hypothèse de récurrence, g est représentable par un arbre de décision ne contenant pas les variables $b_0, \dots, b_i$. Ainsi, g est représentable par un arbre de décision ne contenant pas les variables $b_0, \dots, b_{i-1}$.
- Sinon, les fonctions $g_{i\leftarrow 0}$ et $g_{i\leftarrow 1}$ ne dépendent pas de la variable b_i donc $g_{i\leftarrow 0}, g_{i\leftarrow 1} \in G_{i+1}$. D'après l'hypothèse de récurrence, $g_{i\leftarrow 0}$ est représentable par un arbre de décision f et $g_{i\leftarrow 1}$ est représentable par un arbre de décision v. On a $g_{i\leftarrow 0} \neq g_{i\leftarrow 1}$ donc $v \neq f$ et f et v ne contiennent pas les variables $b_0, \dots, b_i$. Ainsi, l'arbre $A = \mathit{test}_{b_i}(v, f)$ est un arbre de décision ne contenant pas les variables $b_0, \dots, b_{i-1}$.

  Il reste à montrer que A représente la fonction g. Par définition, l'arbre A représente la fonction $\mathit{test}(\pi_i, g_{i\leftarrow 1}, g_{i\leftarrow 0})$. Or, pour tout $b_0, \dots, b_{i-1}, b_{i+1}, \dots, b_{n-1} \in \mathcal{B}$:

  $$\begin{aligned} \mathit{test}(\pi_i, g_{i\leftarrow 1}, g_{i\leftarrow 0})(b_0, \dots, b_{i-1}, 0, b_{i+1}, \dots, b_{n-1}) &= g_{i\leftarrow 0}(b_0, \dots, b_{i-1}, 0, b_{i+1}, \dots, b_{n-1}) \\ &= g(b_0, \dots, b_{i-1}, 0, b_{i+1}, \dots, b_{n-1}), \\ \mathit{test}(\pi_i, g_{i\leftarrow 1}, g_{i\leftarrow 0})(b_0, \dots, b_{i-1}, 1, b_{i+1}, \dots, b_{n-1}) &= g_{i\leftarrow 1}(b_0, \dots, b_{i-1}, 1, b_{i+1}, \dots, b_{n-1}) \\ &= g(b_0, \dots, b_{i-1}, 1, b_{i+1}, \dots, b_{n-1}). \end{aligned}$$

  Donc l'arbre A représente bien la fonction g.

Question 4.a – On a :

$$\boxed{g = (\neg\pi_i \wedge g_{i\leftarrow 0}) \vee (\pi_i \wedge g_{i\leftarrow 1}) \qquad \text{et} \qquad \neg g = (\neg\pi_i \wedge \neg g_{i\leftarrow 0}) \vee (\pi_i \wedge \neg g_{i\leftarrow 1})}$$

Justifions l'égalité $g = (\neg\pi_i \wedge g_{i\leftarrow 0}) \vee (\pi_i \wedge g_{i\leftarrow 1})$. Soit $b = (b_0, \dots, b_{n-1}) \in \mathcal{B}^n$ quelconques. Si $b_i = 0$ alors :

$$\begin{aligned} \Big[(\neg\pi_i \wedge g_{i\leftarrow 0}) \vee (\pi_i \wedge g_{i\leftarrow 1})\Big](b) &= (\neg\pi_i(b) \wedge g_{i\leftarrow 0}(b)) \vee (\pi_i(b) \wedge g_{i\leftarrow 1}(b)) \\ &= (1 \wedge g_{i\leftarrow 0}(b)) \vee (0 \wedge g_{i\leftarrow 1}(b)) \\ &= g_{i\leftarrow 0}(b) \\ &= g_{i\leftarrow 0}(b_0, \dots, b_{i-1}, 0, b_{i+1}, \dots, b_{n-1}) \\ &= g(b_0, \dots, b_{i-1}, 0, b_{i+1}, \dots, b_{n-1}) \\ &= g(b). \end{aligned}$$

De même si $b_i = 1$.
L'égalité $\neg g = (\neg\pi_i \wedge \neg g_{i\leftarrow 0}) \vee (\pi_i \wedge \neg g_{i\leftarrow 1})$ se justifie de manière similaire.

Question 4.b – Montrons que :

$$\boxed{test(g, g', g'') = test\Big(\pi_i, test(g_{i\leftarrow 1}, g'_{i\leftarrow 1}, g''_{i\leftarrow 1}), test(g_{i\leftarrow 0}, g'_{i\leftarrow 0}, g''_{i\leftarrow 0})\Big).}$$

Soit $h : \mathcal{B}^n \to \mathcal{B}$ la fonction définie par l'égalité $h = (g \wedge g') \vee (\neg g \wedge g'')$. Montrons que $h = test(g, g', g'')$. Soit $b \in \mathcal{B}^n$. Si $g(b) = 0$ alors :

$$h(b) = (0 \wedge g'(b)) \vee (1 \wedge g''(b)) = g''(b)$$

Sinon, $g(b) = 1$ donc :

$$h(b) = (1 \wedge g'(b)) \vee (0 \wedge g''(b)) = g'(b)$$

En comparant avec la définition de $test(g, g', g'')$, on conclut que $h = test(g, g', g'')$, c'est à dire :

$$test(g, g', g'') = (g \wedge g') \vee (\neg g \wedge g'').$$

D'après la question précédente :

$$\begin{aligned} g \wedge g' &= \Big[(\neg\pi_i \wedge g_{i\leftarrow 0}) \vee (\pi_i \wedge g_{i\leftarrow 1})\Big] \wedge \Big[(\neg\pi_i \wedge g'_{i\leftarrow 0}) \vee (\pi_i \wedge g'_{i\leftarrow 1})\Big] \\ &= (\neg\pi_i \wedge g_{i\leftarrow 0} \wedge g'_{i\leftarrow 0}) \vee (\pi_i \wedge g_{i\leftarrow 1} \wedge g'_{i\leftarrow 1}) \end{aligned}$$

et :

$$\begin{aligned} \neg g \wedge g'' &= \Big[(\neg\pi_i \wedge \neg g_{i\leftarrow 0}) \vee (\pi_i \wedge \neg g_{i\leftarrow 1})\Big] \wedge \Big[(\neg\pi_i \wedge g''_{i\leftarrow 0}) \vee (\pi_i \wedge g''_{i\leftarrow 1})\Big] \\ &= (\neg\pi_i \wedge \neg g_{i\leftarrow 0} \wedge g''_{i\leftarrow 0}) \vee (\pi_i \wedge \neg g_{i\leftarrow 1} \wedge g''_{i\leftarrow 1}) \end{aligned}$$

Finalement :

$$\begin{aligned} test(g, g', g'') &= (\neg\pi_i \wedge g_{i\leftarrow 0} \wedge g'_{i\leftarrow 0}) \vee (\pi_i \wedge g_{i\leftarrow 1} \wedge g'_{i\leftarrow 1}) \\ &\qquad \vee (\neg\pi_i \wedge \neg g_{i\leftarrow 0} \wedge g''_{i\leftarrow 0}) \vee (\pi_i \wedge \neg g_{i\leftarrow 1} \wedge g''_{i\leftarrow 1}) \\ &= \Big[\pi_i \wedge \Big((g_{i\leftarrow 1} \wedge g'_{i\leftarrow 1}) \vee (\neg g_{i\leftarrow 1} \wedge g''_{i\leftarrow 1})\Big)\Big] \\ &\qquad \vee \Big[\neg\pi_i \wedge \Big((g_{i\leftarrow 0} \wedge g'_{i\leftarrow 0}) \vee (\neg g_{i\leftarrow 0} \wedge g''_{i\leftarrow 0})\Big)\Big] \\ &= \Big[\pi_i \wedge test(g_{i\leftarrow 1}, g'_{i\leftarrow 1}, g''_{i\leftarrow 1})\Big] \vee \Big[\neg\pi_i \wedge test(g_{i\leftarrow 0}, g'_{i\leftarrow 0}, g''_{i\leftarrow 0})\Big] \\ &= test\Big(\pi_i, test(g_{i\leftarrow 1}, g'_{i\leftarrow 1}, g''_{i\leftarrow 1}), test(g_{i\leftarrow 0}, g'_{i\leftarrow 0}, g''_{i\leftarrow 0})\Big) \end{aligned}$$

Question 5 –

```
let abd_proj i = Test(i, Bool true, Bool false);;
```

Question 6.a –

```
let rec abd_neg = function
  | Bool b -> Bool (not b)
  | Test (i, v, f) -> Test (i, abd_neg v, abd_neg f);;
```

Question 6.b – Tout d'abord, la fonction `abd_neg` est injective. Pour le prouver, on utilise une récurrence forte sur la hauteur d'un arbre `a` pour montrer que pour tout arbre `b`, si `(abd_neg a)` est égal à `(abd_neg b)` alors `a` et `b` sont égaux.

Ensuite, par une récurrence forte sur la hauteur de l'arbre `a`, on montre que `a` et `(abd_neg a)` contiennent les mêmes variables.

Finalement, on peut montrer la proposition par disjonction de cas. On note a l'arbre en entrée et a' l'arbre en sortie :

- Si a est réduit à une feuille alors a' est un arbre réduit à une feuille. Donc a' est un arbre de décision bien formé.
- Sinon $a = test_{b_i}(v, f)$ avec v et f des arbres vérifiant les conditions (C 2.1) et (C 2.2). On note v' et f' les arbres renvoyés par la fonction `abd_neg` lorsqu'elle est appliquée aux arbres v et f. On a $a' = test_{b_i}(v', f')$. D'après ce qui précède :
  - Puisque $v \neq f$, on a $v' \neq f'$ donc la condition (C 2.1) est respectée pour a'.
  - Les arbres v et v' contiennent les mêmes variables ainsi que f et f'. Donc la condition (C 2.2) est respectée pour a'.

  Finalement a' est un arbre de décision bien formé.

Question 6.c – On note a_1 l'arbre en entrée de la fonction `abd_neg` et a_2 l'arbre en sortie. On note également g_1 et g_2 les fonctions représentées par les arbres a_1 et a_2. Montrons par récurrence forte sur la hauteur h de a_1 que $g_2 = \neg g_1$:

- Si $h = 0$ alors l'arbre a_1 est réduit à une feuille. Si a_1 vaut $\perp$ alors a_2 vaut $\top$. Ainsi, g_1 est la fonction constante égale à 0 et g_2 est la fonction constante égale à 1. On obtient bien l'égalité $g_2 = \neg g_1$. Idem si a_1 vaut $\top$.
- Soit $h \in \mathbb{N}^*$. On suppose que `abd_neg` est correcte pour tout arbre dont la hauteur est strictement inférieure à h et on le montre pour un arbre de hauteur h. Un arbre a_1 de hauteur $h \geqslant 1$ est de la forme $a_1 = test_{b_i}(v_1, f_1)$ avec v_1 et f_1 des arbres de hauteur strictement inférieure à h. On note v_2 et f_2 les arbres renvoyés par la fonction `abd_neg` lorsqu'elle est appliquée aux arbres v_1 et f_1. On a $a_2 = test_{b_i}(v_2, f_2)$. On note g_1', g_1'', g_2' et g_2'' les fonctions représentées par les arbres v_1, f_1, v_2 et f_2. D'après l'hypothèse de récurrence :

$$g_2' = \neg g_1' \qquad \text{et} \qquad g_2'' = \neg g_1''.$$

  On a donc :

$$g_2 = test(\pi_i, g_2', g_2'') = test(\pi_i, \neg g_1', \neg g_1'') = \neg test(\pi_i, g_1', g_1'') = \neg g_1$$

Question 7 –

```
let rec abd_egal a1 a2 = match a1, a2 with
  | Bool true, Bool true -> true
  | Bool false, Bool false -> true
  | Test(i1, v1, f1), Test(i2, v2, f2) ->
     i1 = i2 && abd_egal v1 v2 && abd_egal f1 f2
  | _ -> false;;
```

Question 8 –

```
let rec abd_partiel i0 b0 = function
  | Test(i, v, f) when i0 = i -> if b0 then v else f
  | Test(i, v, f) when i0 > i ->
     let v1 = abd_partiel i0 b0 v in
     let f1 = abd_partiel i0 b0 f in
     if abd_egal v1 f1 then v1 else Test(i, v1, f1)
  | a -> a;;
```

Question 9 –

```
(* On utilise la question 4.b avec i le plus petit indice d'une
   variable qui apparait dans les arbres c, v et f. *)
let var_racine = function
  | Test(i, _, _) -> i
  | Bool _ -> max_int;;
let var_min c v f = min (var_racine c) (min (var_racine v) (var_racine f));;
let rec abd_test c v f = match c with
  | Bool true  -> v
  | Bool false -> f
  | _ ->
     let i = var_min c v f in
     let v_res =
       let fct = abd_partiel i true in
       abd_test (fct c) (fct v) (fct f) in
     let f_res =
       let fct = abd_partiel i false in
       abd_test (fct c) (fct v) (fct f) in
     if abd_egal v_res f_res then v_res else Test(i, v_res, f_res);;
```

Question 10 –

```
let abd_et a1 a2 = abd_test a1 a2 (Bool false);;
let abd_ou a1 a2 = abd_test a1 (Bool true) a2;;
let abd_implique a1 a2 = abd_test a1 a2 (Bool true);;
```