Principes de fonctionnement des machines binaires

2019/2020

Pierluigi Crescenzi
Université de Paris, IRIF

Université de Paris

INSTITUT DE RECHERCHE EN INFORMATIQUE FONDAMENTALE

cnrs dépasser les frontières

- Tests et examens
  - CC : résultat des tests en TD / TP (semaine 4 et semaine 10)
  - E0 : partiel (samedi 26 octobre)
  - E1 : examen mi décembre
  - E2 : examen fin juin
- Notes finales
  - Note session 1 : 25% CC + 25% E0 + 50% E1
  - Note session 2 : max(E2, 33% CC + 67% E2)
- Rappel
  - Pas de note $\Rightarrow$ pas de moyenne $\Rightarrow$ pas de semestre
- Site web
  - `moodlesupd.script.univ-paris-diderot.fr`

- Numération et arithmétique
- Numération et arithmétique en machine
- Numérisation et codage (texte, images)
- Compression, cryptographie, contrôle d'erreur
- **Logique et calcul propositionnel**
- Circuits numériques

Logique et calcul propositionnel

- Le **calcul propositionnel** ou calcul des propositions est une théorie logique qui formalise le raisonnement logique

- Le **calcul propositionnel** ou calcul des propositions est une théorie logique qui formalise le raisonnement logique
  - Exemple : le fameux syllogisme de Socrate

- Le **calcul propositionnel** ou calcul des propositions est une théorie logique qui formalise le raisonnement logique
  - Exemple : le fameux syllogisme de Socrate
    - Si *Tous les hommes sont mortels*...
    - ...et *Socrate est un homme*,

- Le **calcul propositionnel** ou calcul des propositions est une théorie logique qui formalise le raisonnement logique
  - Exemple : le fameux syllogisme de Socrate
    - Si *Tous les hommes sont mortels*...
    - ...et *Socrate est un homme,*
    - alors *Socrate est mortel*
    - 2 prémisses qui mènent à 1 conclusion

- Le **calcul propositionnel** ou calcul des propositions est une théorie logique qui formalise le raisonnement logique
  - Exemple : le fameux syllogisme de Socrate
    - Si *Tous les hommes sont mortels*...
    - ...et *Socrate est un homme,*
    - alors *Socrate est mortel*
    - 2 prémisses qui mènent à 1 conclusion
- Le calcul propositionnel ne se préoccupe que de la bonne articulation logique des propositions entre elles pas de leur vérité intrinsèque

- Le **calcul propositionnel** ou calcul des propositions est une théorie logique qui formalise le raisonnement logique
  - Exemple : le fameux syllogisme de Socrate
    - Si *Tous les hommes sont mortels*...
    - ...et *Socrate est un homme,*
    - alors *Socrate est mortel*
    - 2 prémisses qui mènent à 1 conclusion
- Le calcul propositionnel ne se préoccupe que de la bonne articulation logique des propositions entre elles pas de leur vérité intrinsèque
  - Exemple : un calcul propositionnel invalide
    - Si *Tous les chats ont des moustaches*..
    - ...et *Mon grand-père a des moustaches,*

- Le **calcul propositionnel** ou calcul des propositions est une théorie logique qui formalise le raisonnement logique
  - Exemple : le fameux syllogisme de Socrate
    - Si *Tous les hommes sont mortels*...
    - ...et *Socrate est un homme*,
    - alors *Socrate est mortel*
    - 2 prémisses qui mènent à 1 conclusion
- Le calcul propositionnel ne se préoccupe que de la bonne articulation logique des propositions entre elles pas de leur vérité intrinsèque
  - Exemple : un calcul propositionnel invalide
    - Si *Tous les chats ont des moustaches*..
    - ...et *Mon grand-père a des moustaches*,
    - alors *Mon grand-père est un chat*

- L'**algèbre de Boole** est le calcul des propositions dont le domaine est celui des booléens
  - Ensemble a 2 éléments {FAUX, VRAI}, $\{\texttt{false}, \texttt{true}\}$, $\{\top, \bot\}$ ou {0, 1} qu'on appelle **valeurs de vérité**

- Une **proposition** est une entité qui donne une information à propos d’un état des choses

- Une **proposition** est une entité qui donne une information à propos d'un état des choses
- Certaines sont universellement vraies ou fausses (ce sont des **constantes**)
  - 12 est un nombre paire
  - 12 est un nombre premier

- Une **proposition** est une entité qui donne une information à propos d'un état des choses
- Certaines sont universellement vraies ou fausses (ce sont des **constantes**)
    - 12 est un nombre paire
    - 12 est un nombre premier
- Certaines sont parfois vraies parfois fausses selon le contexte
    - Il pleut
    - Nous sommes 208 dans cet amphi
    - Aujourd'hui, c'est dimanche
    - La terre est ronde
    - Les tigres sont carnivores

- Une **proposition** est une entité qui donne une information à propos d'un état des choses
- Certaines sont universellement vraies ou fausses (ce sont des **constantes**)
  - 12 est un nombre paire
  - 12 est un nombre premier
- Certaines sont parfois vraies parfois fausses selon le contexte
  - Il pleut
  - Nous sommes 208 dans cet amphi
  - Aujourd'hui, c'est dimanche
  - La terre est ronde
  - Les tigres sont carnivores
- Les propositions sont dénommées à l'aide de symboles
  - **Variables propositionnelles** : $p, q, r, s, \ldots$
  - Ce sont les premiers constituants du langage

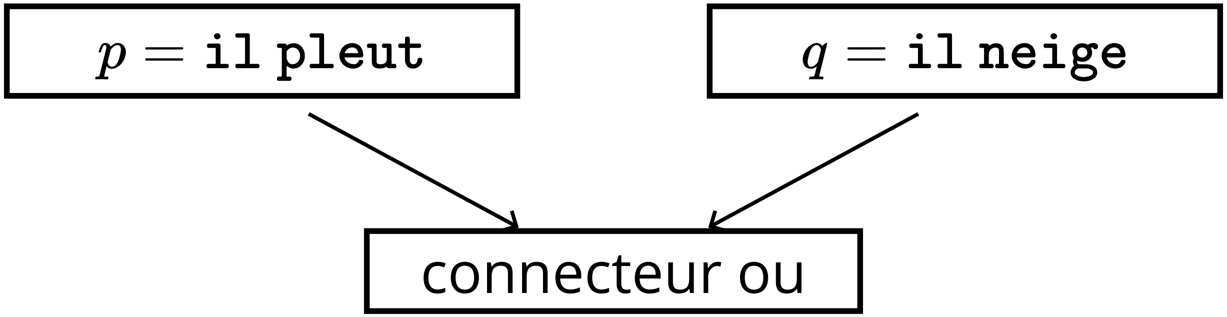
- Les seconds constituants sont les **connecteurs** (ou opérateurs)
  - Permettent de construire des propositions plus élaborées

- Les seconds constituants sont les **connecteurs** (ou opérateurs)
  - Permettent de construire des propositions plus élaborées
- Un exemple de connecteur binaire est le **ou**
  - Permet de combiner deux propositions pour en obtenir une plus complexe dont la valeur de vérité est fonction de la valeur des propositions combinée

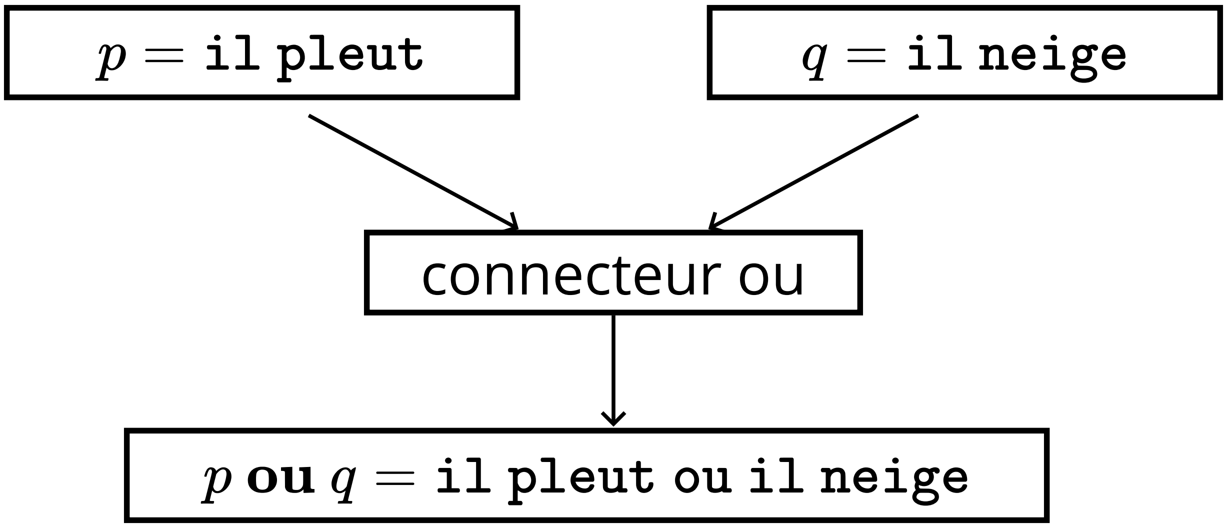
- Les seconds constituants sont les **connecteurs** (ou opérateurs)
  - Permettent de construire des propositions plus élaborées
- Un exemple de connecteur binaire est le **ou**
  - Permet de combiner deux propositions pour en obtenir une plus complexe dont la valeur de vérité est fonction de la valeur des propositions combinée

$\boxed{p = \texttt{il pleut}}$ $\boxed{q = \texttt{il neige}}$

- Les seconds constituants sont les **connecteurs** (ou opérateurs)
  - Permettent de construire des propositions plus élaborées
- Un exemple de connecteur binaire est le **ou**
  - Permet de combiner deux propositions pour en obtenir une plus complexe dont la valeur de vérité est fonction de la valeur des propositions combinée



- Les seconds constituants sont les **connecteurs** (ou opérateurs)
  - Permettent de construire des propositions plus élaborées
- Un exemple de connecteur binaire est le **ou**
  - Permet de combiner deux propositions pour en obtenir une plus complexe dont la valeur de vérité est fonction de la valeur des propositions combinée



- Les connecteurs n-aires sont des applications n-aires de $\{\top, \bot\}^n$ à $\{\top, \bot\}$

- Les connecteurs n-aires sont des applications n-aires de $\{\top, \bot\}^n$ à $\{\top, \bot\}$
  - 2 connecteurs 0-aires : constantes $\top$ et $\bot$
    - Java : `true` et `false`

- Les connecteurs n-aires sont des applications n-aires de $\{\top, \bot\}^n$ à $\{\top, \bot\}$
  - 2 connecteurs 0-aires : constantes $\top$ et $\bot$
    - Java : `true` et `false`
  - 4 connecteurs unaires : $\{\top, \bot\} \to \{\top, \bot\}$
    - Java : !

- Les connecteurs n-aires sont des applications n-aires de $\{\top,\bot\}^n$ à $\{\top,\bot\}$
  - 2 connecteurs 0-aires : constantes $\top$ et $\bot$
    - Java : `true` et `false`
  - 4 connecteurs unaires : $\{\top,\bot\} \to \{\top,\bot\}$
    - Java : !
  - 16 connecteurs binaires : $\{\top,\bot\}^2 \to \{\top,\bot\}$
    - Java : ||, `&&` et ^

- Les connecteurs n-aires sont des applications n-aires de $\{\top,\bot\}^n$ à $\{\top,\bot\}$
  - 2 connecteurs 0-aires : constantes $\top$ et $\bot$
    - Java : `true` et `false`
  - 4 connecteurs unaires : $\{\top,\bot\} \rightarrow \{\top,\bot\}$
    - Java : !
  - 16 connecteurs binaires : $\{\top,\bot\}^2 \rightarrow \{\top,\bot\}$
    - Java : ||, `&&` et ^
  - 256 connecteurs ternaires : $\{\top,\bot\}^3 \rightarrow \{\top,\bot\}$
    - Disjonction conditionnée
  - ...

- Connecteurs unaires

p	absurdisation	identité	négation	tautologisation
$\perp$	$\perp$	$\perp$	$\top$	$\top$
$\top$	$\perp$	$\top$	$\perp$	$\top$

  - La négation de p est souvent notée $\neg p$
    - Java : opérateur !

- Connecteurs binaires

p	q	$\perp$	et	n-imp	p_1	n-pmi	p_2	x-ou	ou
$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$
$\perp$	$\top$	$\perp$	$\perp$	$\perp$	$\perp$	$\top$	$\top$	$\top$	$\top$
$\top$	$\perp$	$\perp$	$\perp$	$\top$	$\top$	$\perp$	$\perp$	$\top$	$\top$
$\top$	$\top$	$\perp$	$\top$	$\perp$	$\top$	$\perp$	$\top$	$\perp$	$\top$

p	q	n-ou	eq	n-p_2	pmi	n-p_1	imp	n-et	$\top$
$\perp$	$\perp$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$
$\perp$	$\top$	$\perp$	$\perp$	$\perp$	$\perp$	$\top$	$\top$	$\top$	$\top$
$\top$	$\perp$	$\perp$	$\perp$	$\top$	$\top$	$\perp$	$\perp$	$\top$	$\top$
$\top$	$\top$	$\perp$	$\top$	$\perp$	$\top$	$\perp$	$\top$	$\perp$	$\top$

- Connecteurs binaires

p	q	$\perp$	et	n-imp	p_1	n-pmi	p_2	x-ou	ou
$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$
$\perp$	$\top$	$\perp$	$\perp$	$\perp$	$\perp$	$\top$	$\top$	$\top$	$\top$
$\top$	$\perp$	$\perp$	$\perp$	$\top$	$\top$	$\perp$	$\perp$	$\top$	$\top$
$\top$	$\top$	$\perp$	$\top$	$\perp$	$\top$	$\perp$	$\top$	$\perp$	$\top$

p	q	n-ou	eq	n-p_2	pmi	n-p_1	imp	n-et	$\top$
$\perp$	$\perp$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$
$\perp$	$\top$	$\perp$	$\perp$	$\perp$	$\perp$	$\top$	$\top$	$\top$	$\top$
$\top$	$\perp$	$\perp$	$\perp$	$\top$	$\top$	$\perp$	$\perp$	$\top$	$\top$
$\top$	$\top$	$\perp$	$\top$	$\perp$	$\top$	$\perp$	$\top$	$\perp$	$\top$

- et : $\wedge$ (Java : `&&` ou `&`)

- Connecteurs binaires

p	q	$\bot$	et	n-imp	p_1	n-pmi	p_2	x-ou	ou
$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$
$\bot$	$\top$	$\bot$	$\bot$	$\bot$	$\bot$	$\top$	$\top$	$\top$	$\top$
$\top$	$\bot$	$\bot$	$\bot$	$\top$	$\top$	$\bot$	$\bot$	$\top$	$\top$
$\top$	$\top$	$\bot$	$\top$	$\bot$	$\top$	$\bot$	$\top$	$\bot$	$\top$

p	q	n-ou	eq	n-p_2	pmi	n-p_1	imp	n-et	$\top$
$\bot$	$\bot$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$
$\bot$	$\top$	$\bot$	$\bot$	$\bot$	$\bot$	$\top$	$\top$	$\top$	$\top$
$\top$	$\bot$	$\bot$	$\bot$	$\top$	$\top$	$\bot$	$\bot$	$\top$	$\top$
$\top$	$\top$	$\bot$	$\top$	$\bot$	$\top$	$\bot$	$\top$	$\bot$	$\top$

- et : $\wedge$ (Java : `&&` ou `&`)
- ou : $\vee$ (Java : `||` ou `|`)

- Connecteurs binaires

p	q	$\bot$	et	n-imp	p_1	n-pmi	p_2	x-ou	ou
$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$
$\bot$	$\top$	$\bot$	$\bot$	$\bot$	$\bot$	$\top$	$\top$	$\top$	$\top$
$\top$	$\bot$	$\bot$	$\bot$	$\top$	$\top$	$\bot$	$\bot$	$\top$	$\top$
$\top$	$\top$	$\bot$	$\top$	$\bot$	$\top$	$\bot$	$\top$	$\bot$	$\top$

p	q	n-ou	eq	n-p_2	pmi	n-p_1	imp	n-et	$\top$
$\bot$	$\bot$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$
$\bot$	$\top$	$\bot$	$\bot$	$\bot$	$\bot$	$\top$	$\top$	$\top$	$\top$
$\top$	$\bot$	$\bot$	$\bot$	$\top$	$\top$	$\bot$	$\bot$	$\top$	$\top$
$\top$	$\top$	$\bot$	$\top$	$\bot$	$\top$	$\bot$	$\top$	$\bot$	$\top$

- et : $\wedge$ (Java : `&&` ou `&`)
- ou : $\vee$ (Java : || ou |)
- x-ou : $\oplus$ (Java : ^)

- Connecteurs binaires

p	q	$\bot$	et	n-imp	p_1	n-pmi	p_2	x-ou	ou
$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$
$\bot$	$\top$	$\bot$	$\bot$	$\bot$	$\bot$	$\top$	$\top$	$\top$	$\top$
$\top$	$\bot$	$\bot$	$\bot$	$\top$	$\top$	$\bot$	$\bot$	$\top$	$\top$
$\top$	$\top$	$\bot$	$\top$	$\bot$	$\top$	$\bot$	$\top$	$\bot$	$\top$

p	q	n-ou	eq	n-p_2	pmi	n-p_1	imp	n-et	$\top$
$\bot$	$\bot$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$
$\bot$	$\top$	$\bot$	$\bot$	$\bot$	$\bot$	$\top$	$\top$	$\top$	$\top$
$\top$	$\bot$	$\bot$	$\bot$	$\top$	$\top$	$\bot$	$\bot$	$\top$	$\top$
$\top$	$\top$	$\bot$	$\top$	$\bot$	$\top$	$\bot$	$\top$	$\bot$	$\top$

- et : $\wedge$ (Java : `&&` ou `&`)
- ou : $\vee$ (Java : || ou |)
- x-ou : $\oplus$ (Java : ^)
- n-et : $\uparrow$

- Connecteurs binaires

p	q	$\perp$	et	n-imp	p_1	n-pmi	p_2	x-ou	ou
$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$
$\perp$	$\top$	$\perp$	$\perp$	$\perp$	$\perp$	$\top$	$\top$	$\top$	$\top$
$\top$	$\perp$	$\perp$	$\perp$	$\top$	$\top$	$\perp$	$\perp$	$\top$	$\top$
$\top$	$\top$	$\perp$	$\top$	$\perp$	$\top$	$\perp$	$\top$	$\perp$	$\top$

p	q	n-ou	eq	n-p_2	pmi	n-p_1	imp	n-et	$\top$
$\perp$	$\perp$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$
$\perp$	$\top$	$\perp$	$\perp$	$\perp$	$\perp$	$\top$	$\top$	$\top$	$\top$
$\top$	$\perp$	$\perp$	$\perp$	$\top$	$\top$	$\perp$	$\perp$	$\top$	$\top$
$\top$	$\top$	$\perp$	$\top$	$\perp$	$\top$	$\perp$	$\top$	$\perp$	$\top$

- et : $\wedge$ (Java : `&&` ou `&`)
- ou : $\vee$ (Java : `||` ou `|`)
- x-ou : $\oplus$ (Java : `^`)
- n-et : $\uparrow$
- n-ou : $\downarrow$

- Connecteurs binaires

p	q	$\perp$	et	n-imp	p_1	n-pmi	p_2	x-ou	ou
$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$
$\perp$	$\top$	$\perp$	$\perp$	$\perp$	$\perp$	$\top$	$\top$	$\top$	$\top$
$\top$	$\perp$	$\perp$	$\perp$	$\top$	$\top$	$\perp$	$\perp$	$\top$	$\top$
$\top$	$\top$	$\perp$	$\top$	$\perp$	$\top$	$\perp$	$\top$	$\perp$	$\top$

p	q	n-ou	eq	n-p_2	pmi	n-p_1	imp	n-et	$\top$
$\perp$	$\perp$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$
$\perp$	$\top$	$\perp$	$\perp$	$\perp$	$\perp$	$\top$	$\top$	$\top$	$\top$
$\top$	$\perp$	$\perp$	$\perp$	$\top$	$\top$	$\perp$	$\perp$	$\top$	$\top$
$\top$	$\top$	$\perp$	$\top$	$\perp$	$\top$	$\perp$	$\top$	$\perp$	$\top$

- et : $\wedge$ (Java : `&&` ou `&`)
- ou : $\vee$ (Java : `||` ou `|`)
- x-ou : $\oplus$ (Java : `^`)
- n-et : $\uparrow$
- n-ou : $\downarrow$
- imp : $\Rightarrow$

- Connecteurs binaires

p	q	$\bot$	et	n-imp	p_1	n-pmi	p_2	x-ou	ou
$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$
$\bot$	$\top$	$\bot$	$\bot$	$\bot$	$\bot$	$\top$	$\top$	$\top$	$\top$
$\top$	$\bot$	$\bot$	$\bot$	$\top$	$\top$	$\bot$	$\bot$	$\top$	$\top$
$\top$	$\top$	$\bot$	$\top$	$\bot$	$\top$	$\bot$	$\top$	$\bot$	$\top$

p	q	n-ou	eq	n-p_2	pmi	n-p_1	imp	n-et	$\top$
$\bot$	$\bot$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$
$\bot$	$\top$	$\bot$	$\bot$	$\bot$	$\bot$	$\top$	$\top$	$\top$	$\top$
$\top$	$\bot$	$\bot$	$\bot$	$\top$	$\top$	$\bot$	$\bot$	$\top$	$\top$
$\top$	$\top$	$\bot$	$\top$	$\bot$	$\top$	$\bot$	$\top$	$\bot$	$\top$

- et : $\wedge$ (Java : `&&` ou `&`)
- ou : $\vee$ (Java : `||` ou `|`)
- x-ou : $\oplus$ (Java : `^`)
- n-et : $\uparrow$
- n-ou : $\downarrow$
- imp : $\Rightarrow$
- eq : $\Leftrightarrow$

- Connecteurs ternaires

p	q	r	si p alors q sinon r
$\perp$	$\perp$	$\perp$	$\perp$
$\perp$	$\perp$	$\top$	$\top$
$\perp$	$\top$	$\perp$	$\perp$
$\perp$	$\top$	$\top$	$\top$
$\top$	$\perp$	$\perp$	$\perp$
$\top$	$\perp$	$\top$	$\perp$
$\top$	$\top$	$\perp$	$\top$
$\top$	$\top$	$\top$	$\top$

- En Java : opérateur ? :
  - Exemple : $\mathtt{s} = \mathtt{p} \mathbin{?} \mathtt{q} : \mathtt{r};$
- $2^8 = 256$ connecteurs ternaires au total

- Les **formules** du calcul propositionnel sont obtenues par induction

- Les **formules** du calcul propositionnel sont obtenues par induction
  - Toute constante est une formule, toute variable est une formule

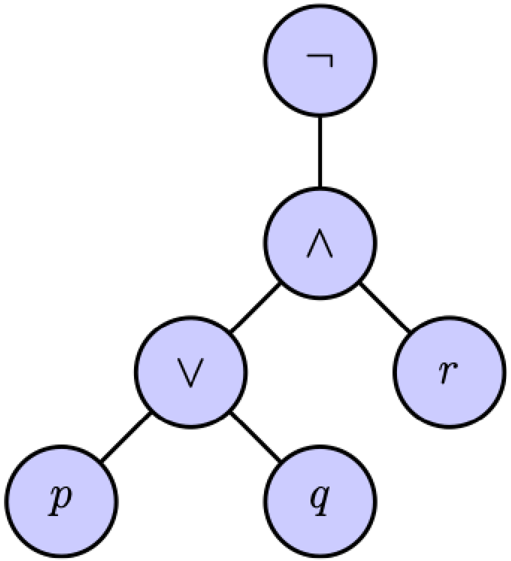
- Les **formules** du calcul propositionnel sont obtenues par induction
  - Toute constante est une formule, toute variable est une formule
  - Pour tout entier n, pour tout connecteur n-aire γ et pour toute formule f_i ($1 \leq i \leq n$), $\gamma(f_1, \ldots, f_n)$ est une formule
    - $n = 2$: $(f_1)\ \gamma\ (f_2)$ (notation infixe)

- Les **formules** du calcul propositionnel sont obtenues par induction
  - Toute constante est une formule, toute variable est une formule
  - Pour tout entier n, pour tout connecteur n-aire γ et pour toute formule f_i $(1 \leq i \leq n)$, $\gamma(f_1, \ldots, f_n)$ est une formule
    - $n = 2$: $(f_1)\ \gamma\ (f_2)$ (notation infixe)
- Exemples
  - $\top$, $\bot$, p, q et r
  - $(p)\ \wedge\ (q)$, $((p)\ \wedge\ (q))\ \vee\ (r)$, $\neg(((p)\ \wedge\ (q))\ \vee\ (r))$

- Les **formules** du calcul propositionnel sont obtenues par induction
  - Toute constante est une formule, toute variable est une formule
  - Pour tout entier n, pour tout connecteur n-aire γ et pour toute formule f_i $(1 \leq i \leq n)$, $\gamma(f_1, \ldots, f_n)$ est une formule
    - $n = 2$: $(f_1)\ \gamma\ (f_2)$ (notation infixe)
- Exemples
  - $\top$, $\bot$, p, q et r
  - $(p) \ \wedge \ (q)$, $((p) \ \wedge \ (q)) \ \vee \ (r)$, $\neg(((p) \ \wedge \ (q)) \ \vee \ (r))$
- Certaines parenthèses peuvent être enlevées (pourvu que des **priorités** soient définies entre opérateurs)

- Les **formules** du calcul propositionnel sont obtenues par induction
  - Toute constante est une formule, toute variable est une formule
  - Pour tout entier n, pour tout connecteur n-aire γ et pour toute formule f_i $(1 \leq i \leq n)$, $\gamma(f_1, \ldots, f_n)$ est une formule
    - $n = 2$: $(f_1)\ \gamma\ (f_2)$ (notation infixe)
- Exemples
  - $\top$, $\bot$, p, q et r
  - $(p) \wedge (q)$, $((p) \wedge (q)) \vee (r)$, $\neg(((p) \wedge (q)) \vee (r))$
- Certaines parenthèses peuvent être enlevées (pourvu que des **priorités** soient définies entre opérateurs)
  - Exemple : $\neg(p \wedge q \vee r)$

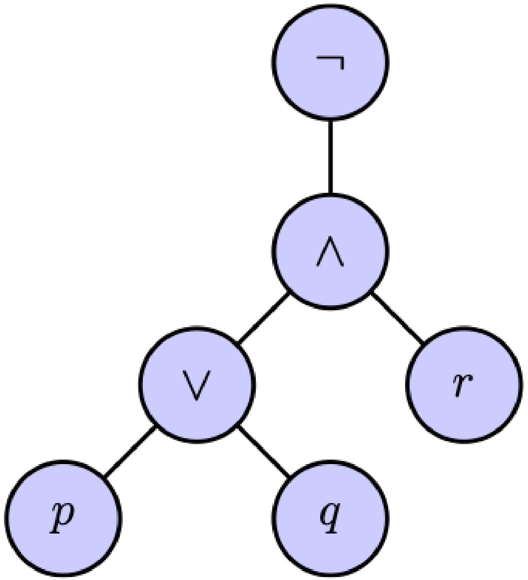
- Il est parfois pratique d'associer à une formule son **arbre syntaxique** qui reflète sa structure profonde
  - Exemple : $\neg((p \lor q) \land r)$



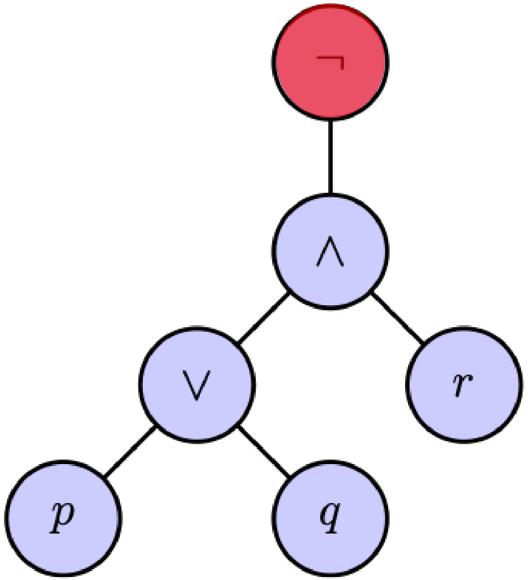
- L'arbre syntaxique exprime la formule indépendamment de sa syntaxe formelle
  - C'est une forme canonique

- La forme standard de l'écriture d'une formule (forme infixe) a l'inconvénient de nécessiter l'usage de parenthèses
  - Il existe deux autres formes que l'on peut extraire facilement de l'arbre syntaxique et qui permettent de s'en passer
- **Forme préfixe** : les opérateurs sont placés avant les opérandes
  - $p \ \wedge \ q : \wedge \, p \, q$
- **Forme suffixe** : les opérateurs sont placés après les opérandes
  - $p \ \wedge \ q : p \, q \, \wedge$

- Forme préfixe et arbre syntaxique

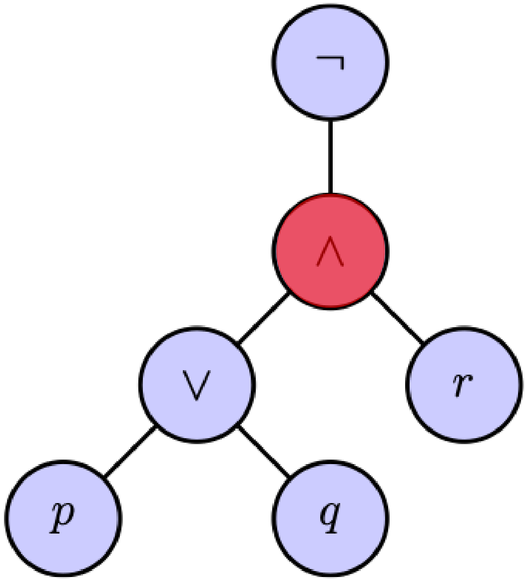


- Forme préfixe et arbre syntaxique



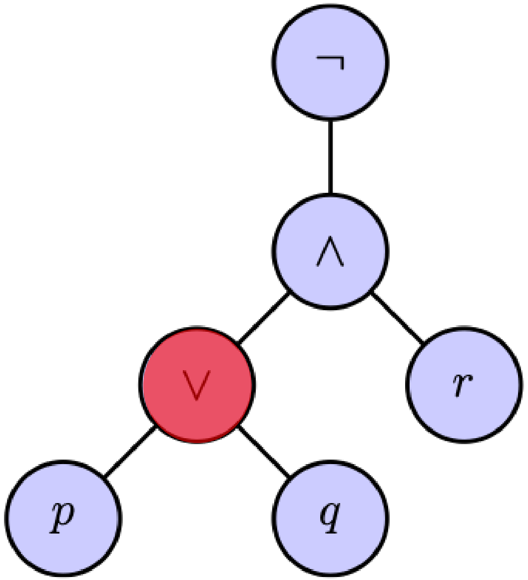
$\neg$

- Forme préfixe et arbre syntaxique



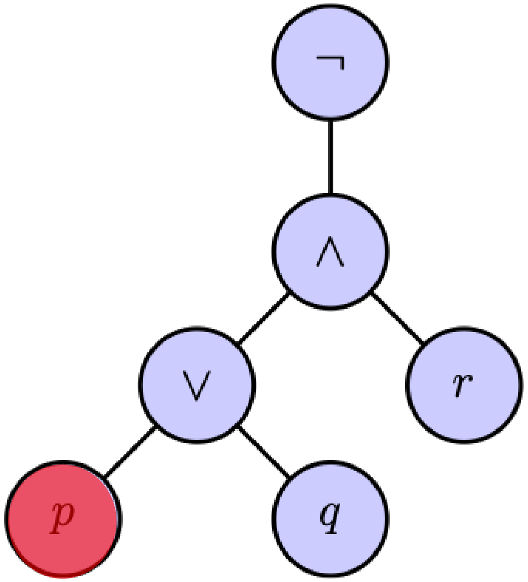
$\neg \quad \wedge$

- Forme préfixe et arbre syntaxique



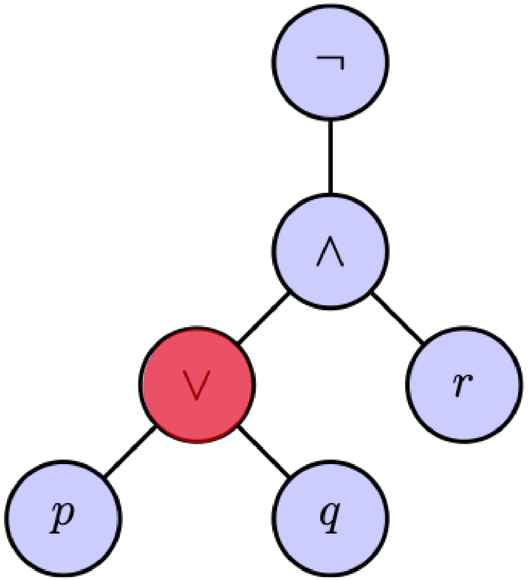
$$\neg \quad \wedge \quad \vee$$

- Forme préfixe et arbre syntaxique



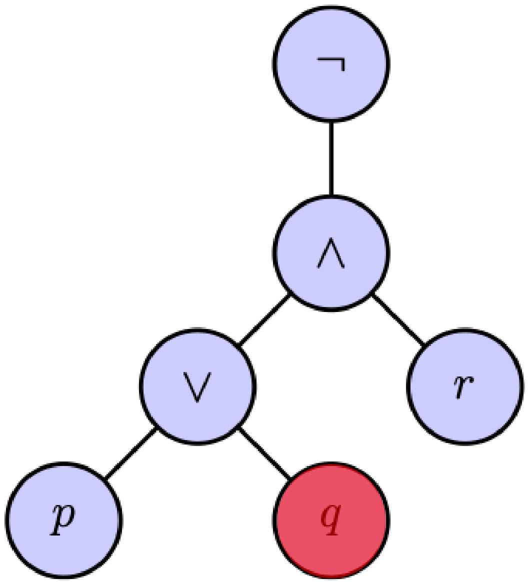
$$\neg \quad \wedge \quad \vee \quad \boldsymbol{p}$$

- Forme préfixe et arbre syntaxique



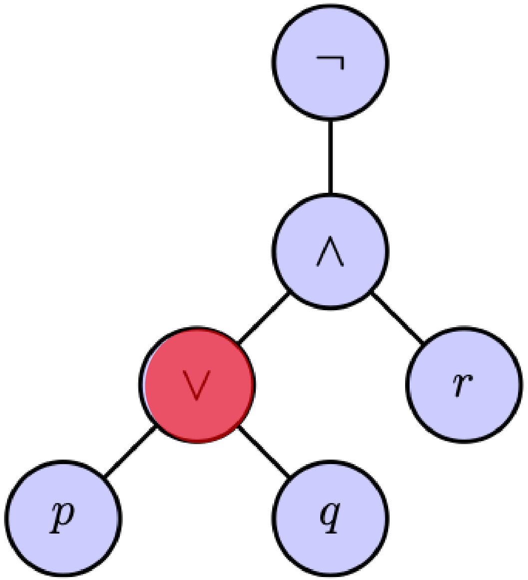
$$\neg \quad \wedge \quad \vee \quad \boldsymbol{p}$$

- Forme préfixe et arbre syntaxique



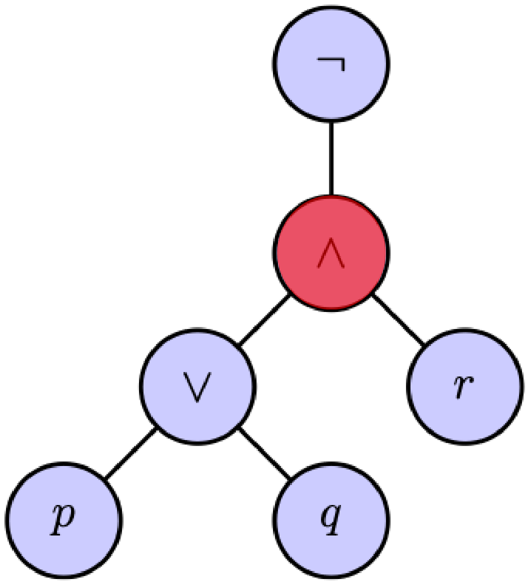
$$\neg \quad \wedge \quad \vee \quad p \quad q$$

- Forme préfixe et arbre syntaxique



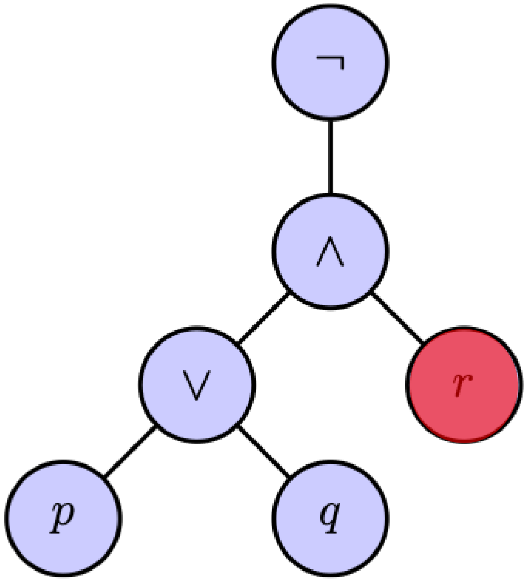
$$\neg \quad \wedge \quad \vee \quad p \quad q$$

- Forme préfixe et arbre syntaxique



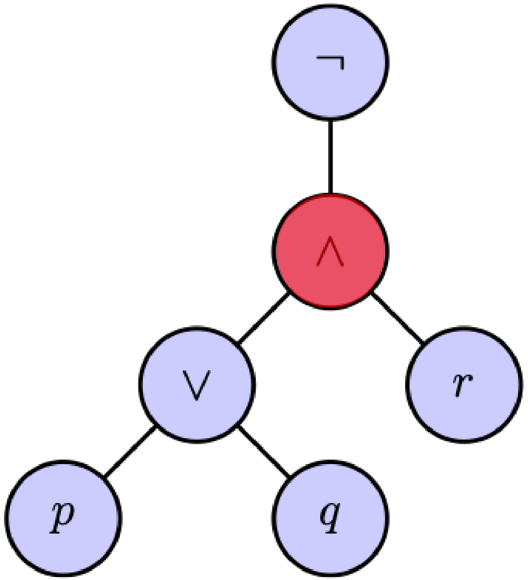
$$\neg \quad \wedge \quad \vee \quad p \quad q$$

- Forme préfixe et arbre syntaxique



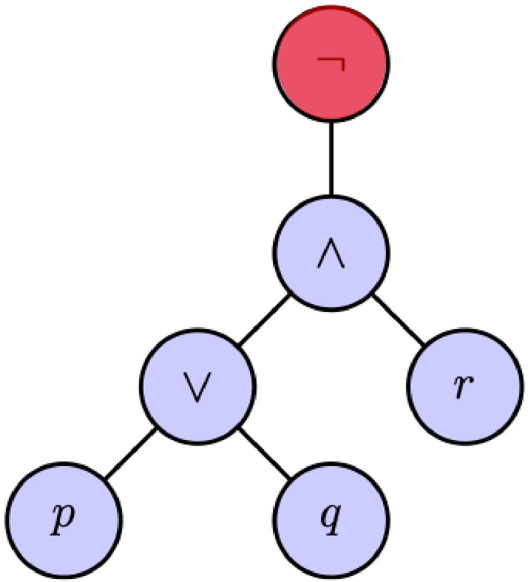
$$\neg \quad \wedge \quad \vee \quad p \quad q \quad r$$

- Forme préfixe et arbre syntaxique



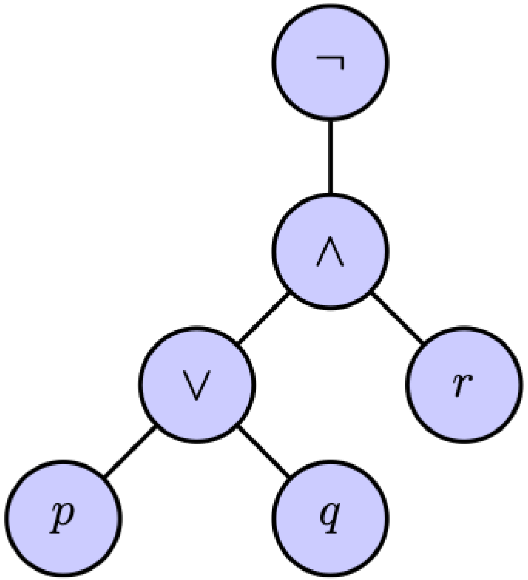
$$\neg \quad \wedge \quad \vee \quad p \quad q \quad r$$

- Forme préfixe et arbre syntaxique

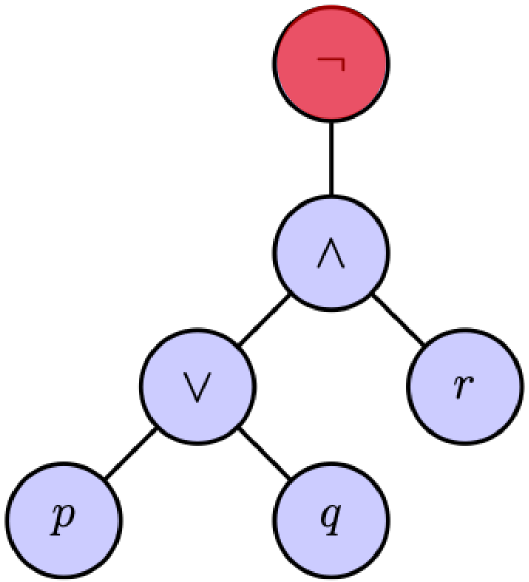


$$\neg \quad \wedge \quad \vee \quad p \quad q \quad r$$

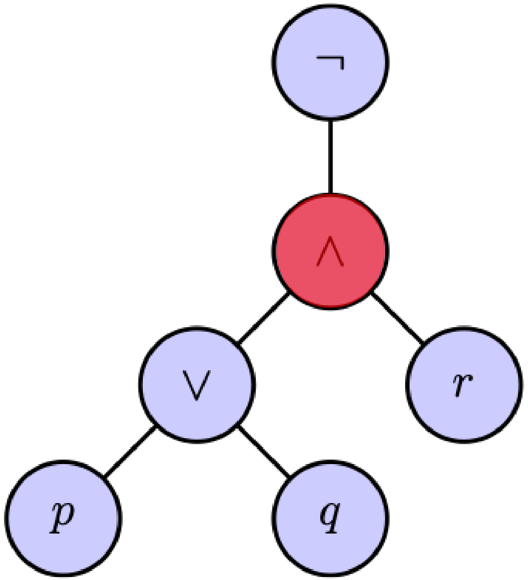
- Forme suffixe et arbre syntaxique



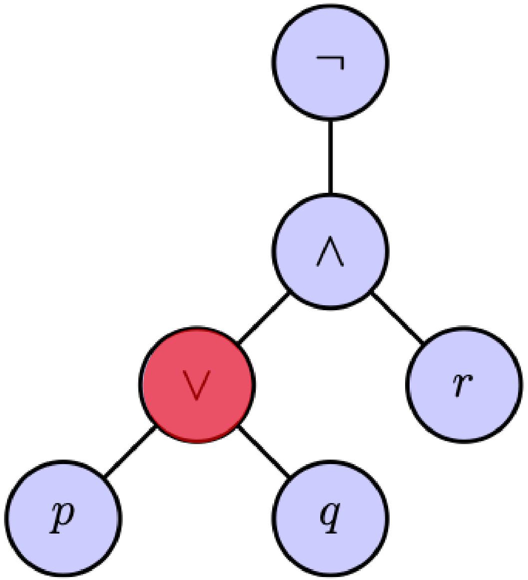
- Forme suffixe et arbre syntaxique



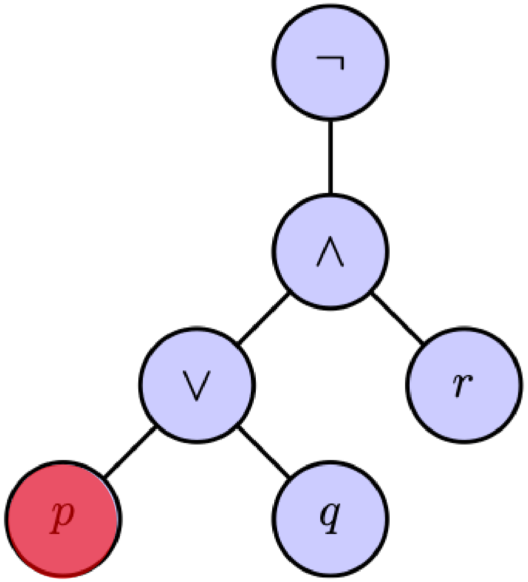
- Forme suffixe et arbre syntaxique



- Forme suffixe et arbre syntaxique

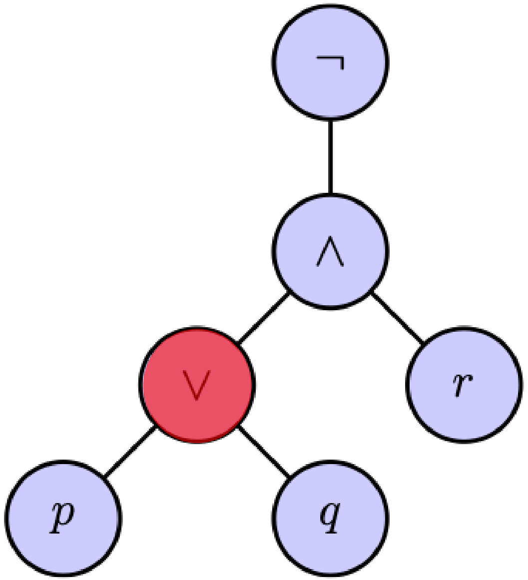


- Forme suffixe et arbre syntaxique



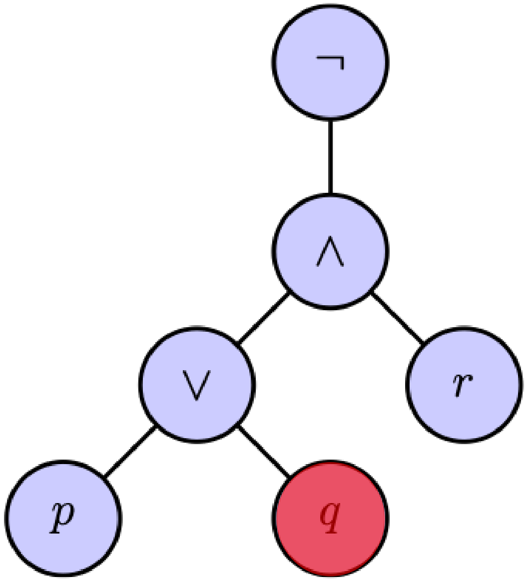
$$p$$

- Forme suffixe et arbre syntaxique



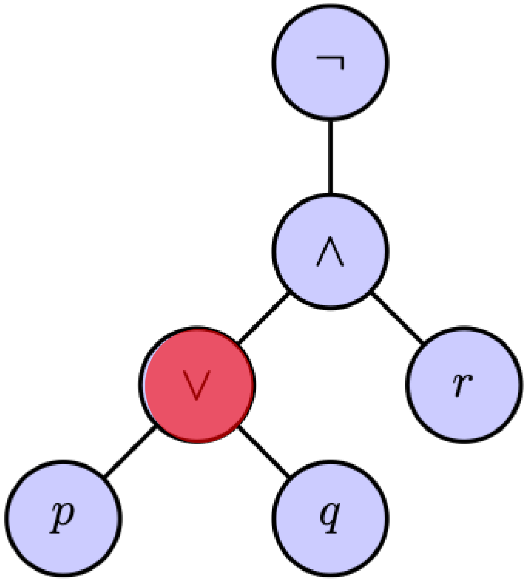
$$p$$

- Forme suffixe et arbre syntaxique



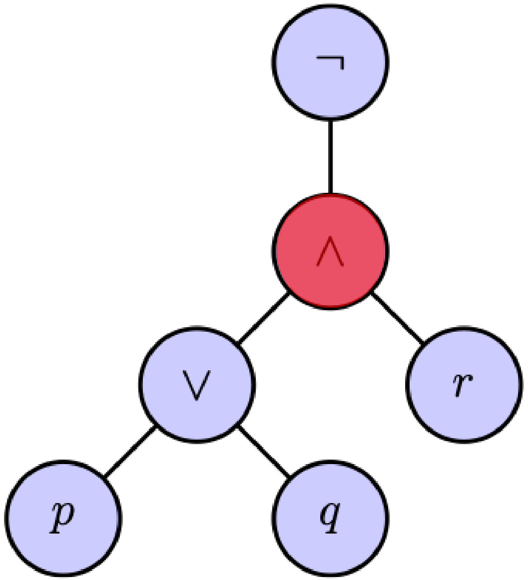
$$p \quad q$$

- Forme suffixe et arbre syntaxique



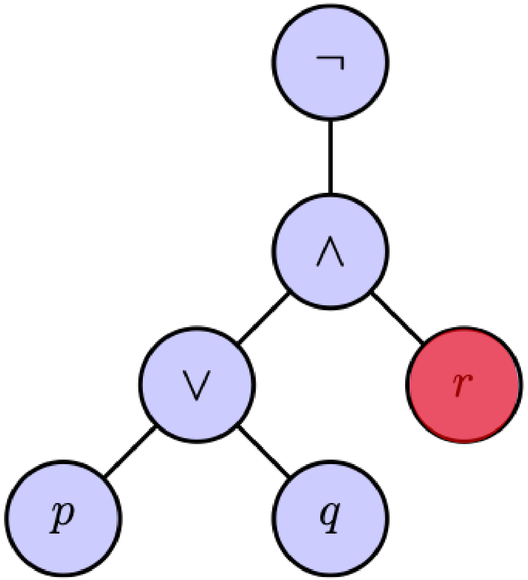
$$p \quad q \quad \vee$$

- Forme suffixe et arbre syntaxique



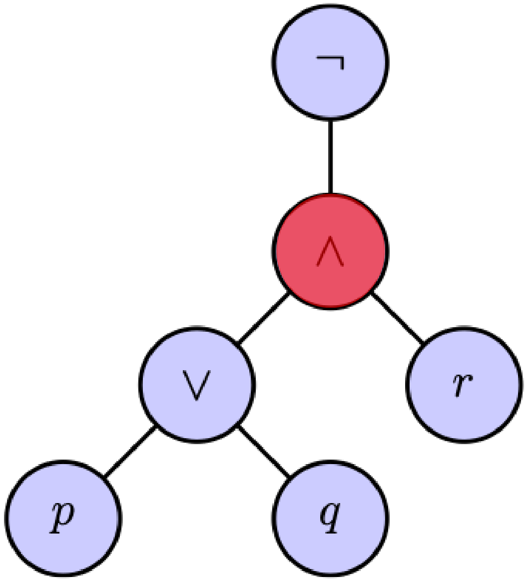
$$p \quad q \quad \vee$$

- Forme suffixe et arbre syntaxique



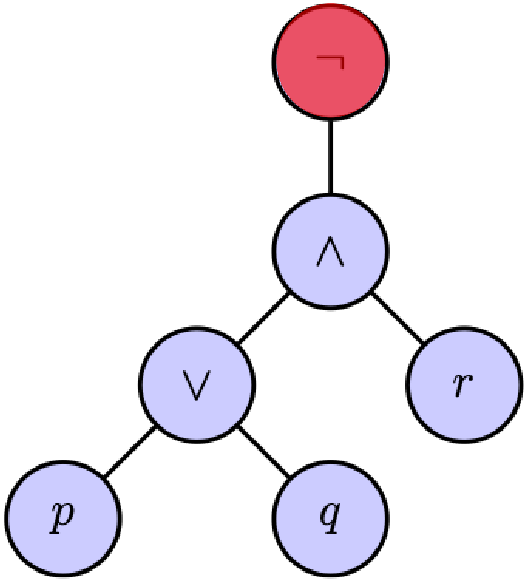
$$p \quad q \quad \vee \quad r$$

- Forme suffixe et arbre syntaxique



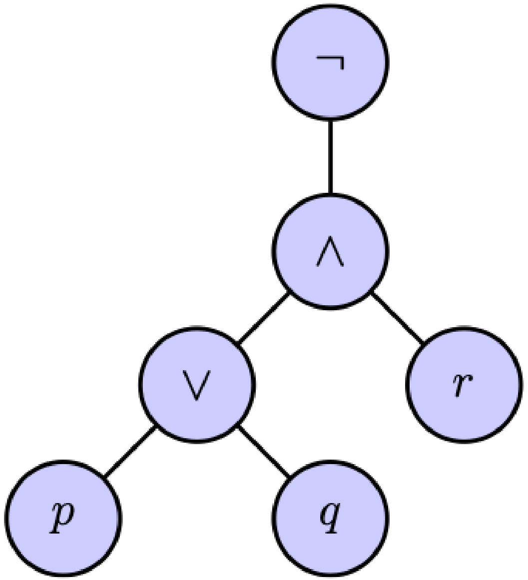
$$p \quad q \quad \vee \quad r \quad \wedge$$

- Forme suffixe et arbre syntaxique

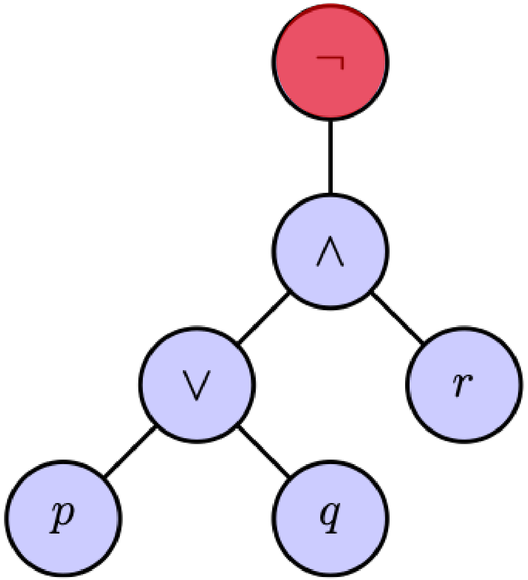


$$p \quad q \quad \vee \quad r \quad \wedge \quad \neg$$

- Forme infixe et arbre syntaxique

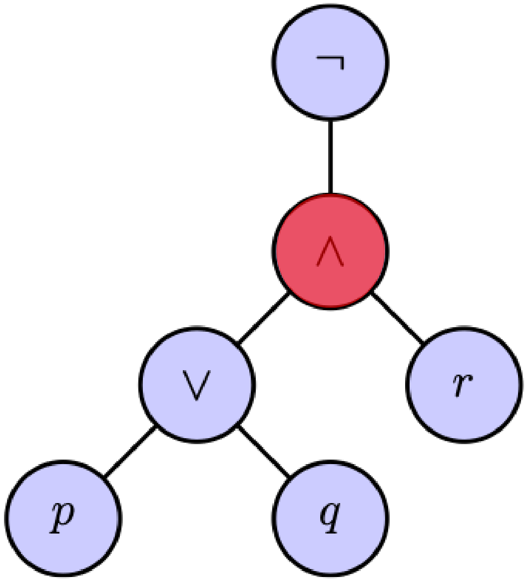


- Forme infixe et arbre syntaxique



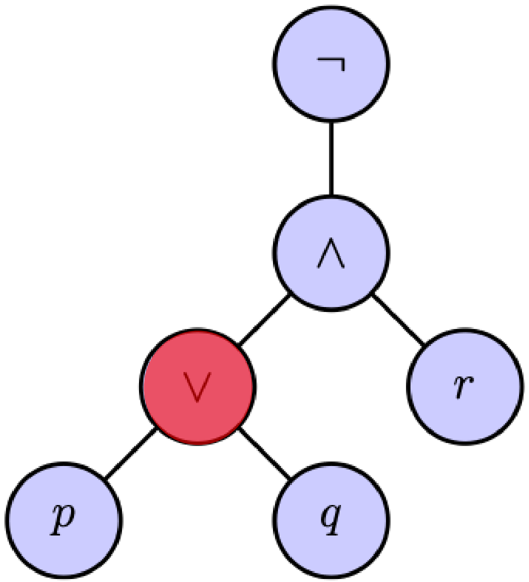
$$\neg \quad ($$

- Forme infixe et arbre syntaxique



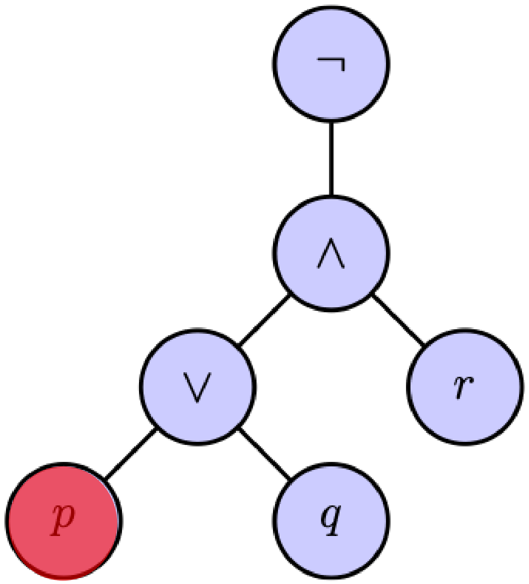
$\neg \quad (\quad ($

- Forme infixe et arbre syntaxique



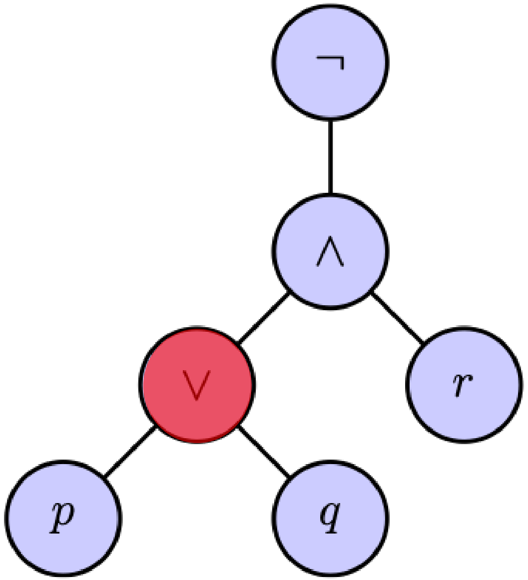
$$\neg \quad (\quad (\quad ($$

- Forme infixe et arbre syntaxique



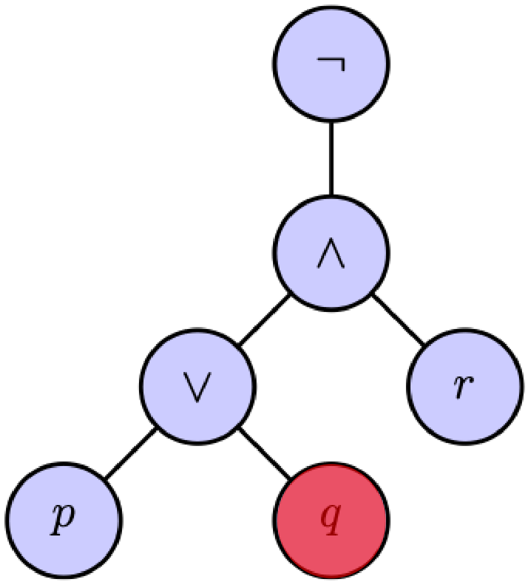
$$\neg \quad (\quad (\quad (\; (p)$$

- Forme infixe et arbre syntaxique



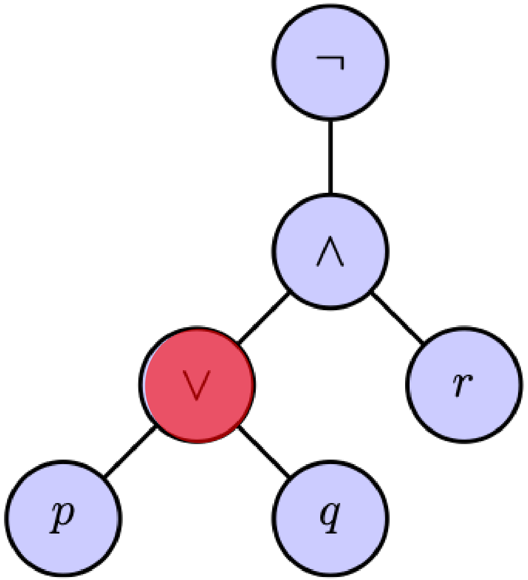
$$\neg \quad (\quad (\quad (\; (p) \vee$$

- Forme infixe et arbre syntaxique



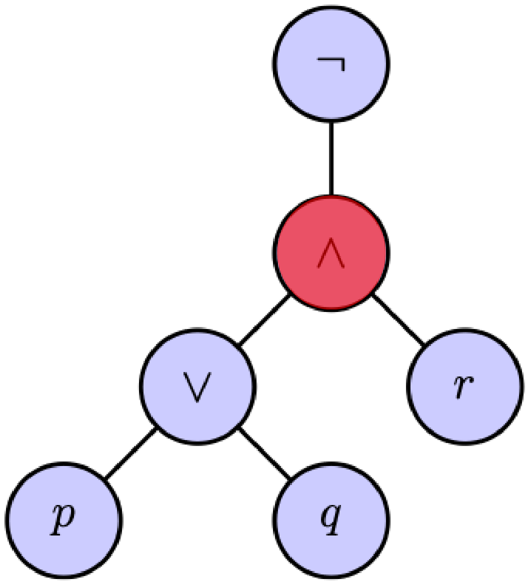
$$\neg \quad (\quad (\quad (\ (p) \vee (q)$$

- Forme infixe et arbre syntaxique



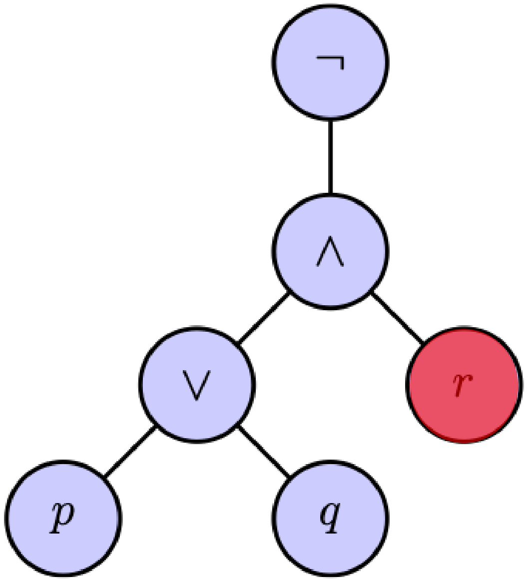
$$\neg \quad (\quad (\quad (\; (p) \vee (q) \;)$$

- Forme infixe et arbre syntaxique



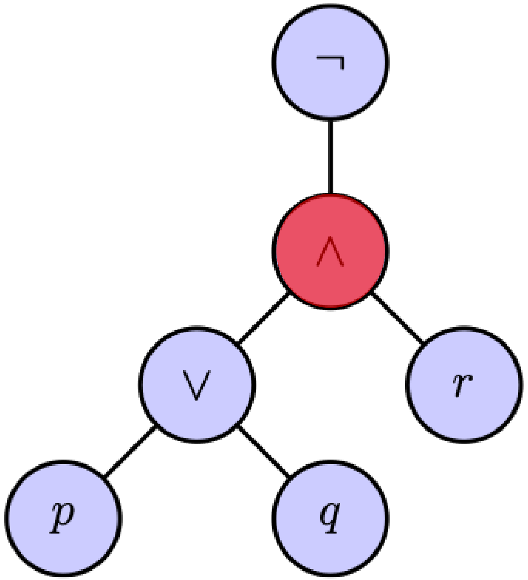
$$\neg \quad (\quad (\quad (\ (p) \vee (q) \ ) \quad \wedge$$

- Forme infixe et arbre syntaxique



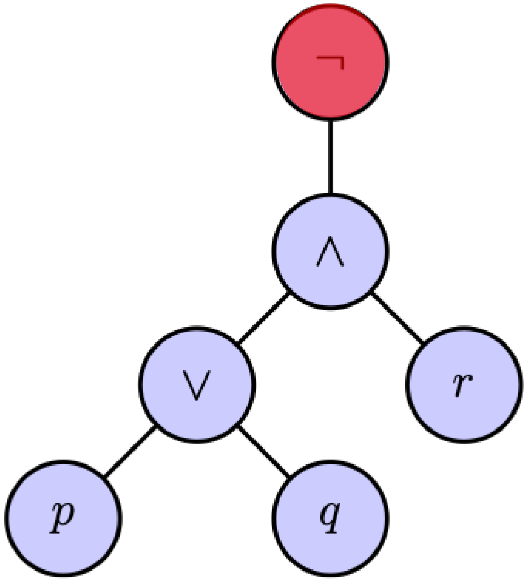
$$\neg \quad (\quad (\quad (\ (p) \vee (q) \ ) \ \wedge (r))$$

- Forme infixe et arbre syntaxique



$$\neg \quad (\quad (\quad (\; (p) \vee (q) \;) \quad \wedge \; (r) \;)$$

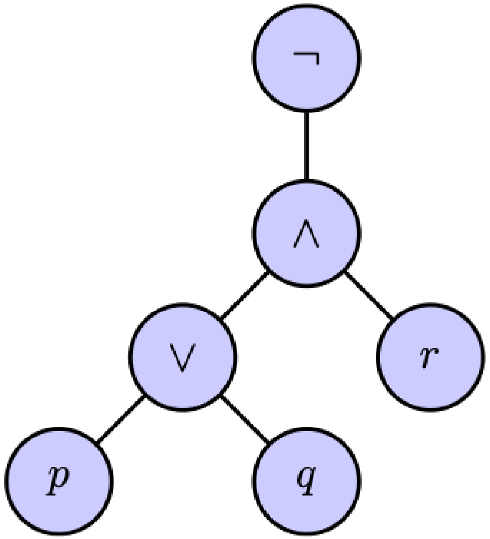
- Forme infixe et arbre syntaxique



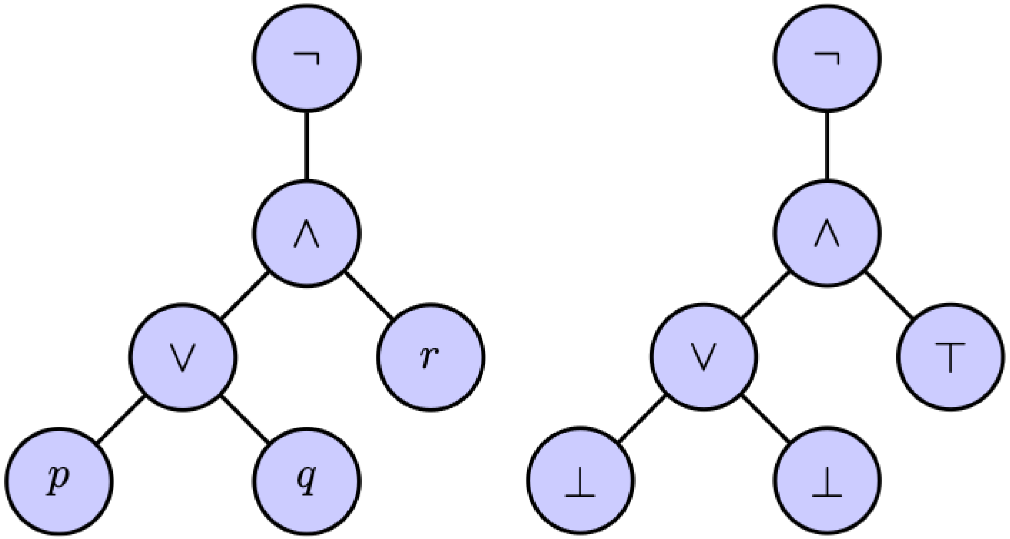
$$\neg\ (\ (\ (\ (p) \vee (q)\ )\ \wedge (r)\ )\ )$$

- L'arbre syntaxique ou arbre de syntaxe abstraite permet aussi d'évaluer une expression étant donnée la valuation des variables

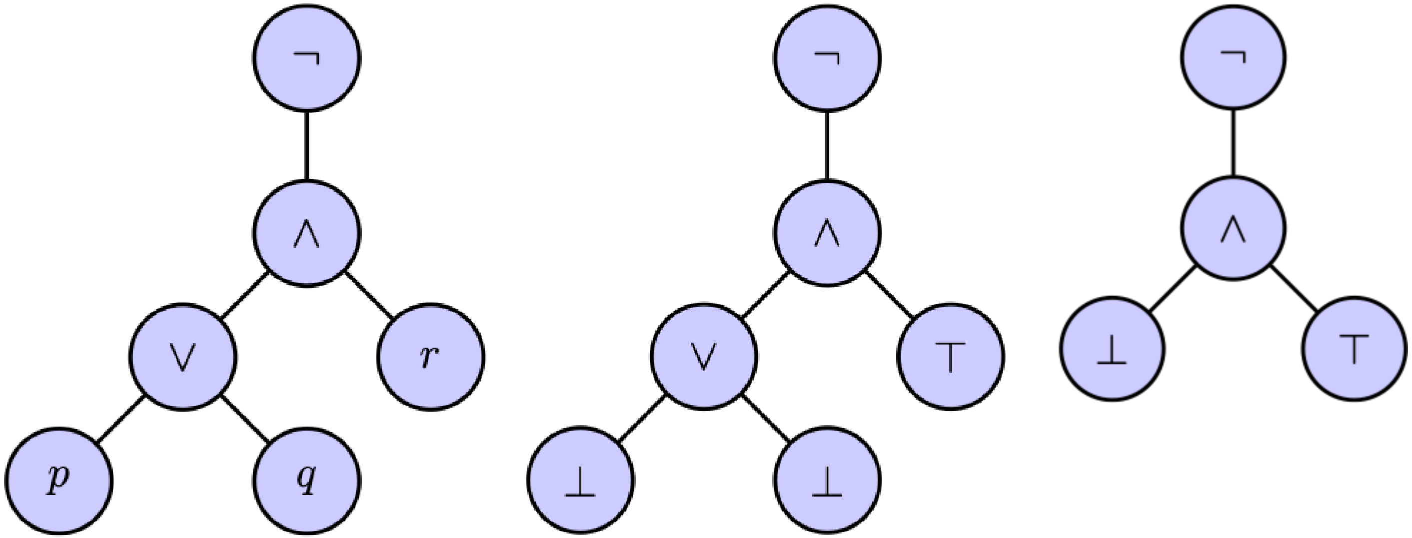
- L'arbre syntaxique ou arbre de syntaxe abstraite permet aussi d'évaluer une expression étant donnée la valuation des variables
  - Exemple : $p = \bot$, $q = \bot$ et $r = \top$



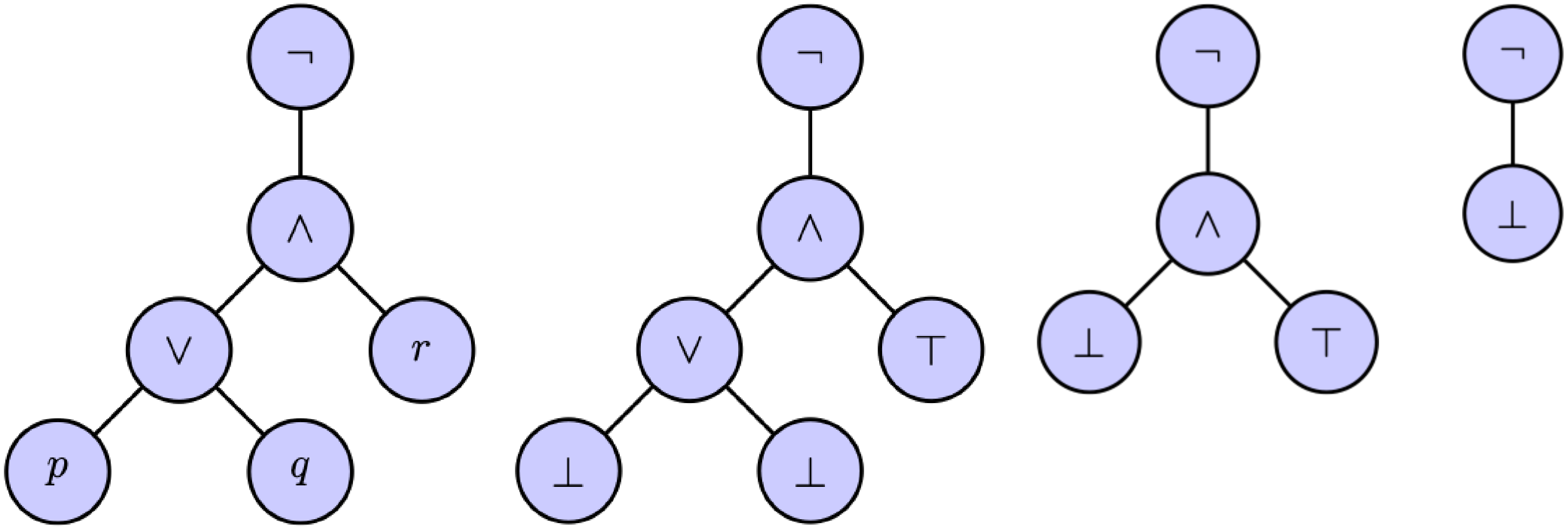
- L'arbre syntaxique ou arbre de syntaxe abstraite permet aussi d'évaluer une expression étant donnée la valuation des variables
  - Exemple : $p = \bot$, $q = \bot$ et $r = \top$



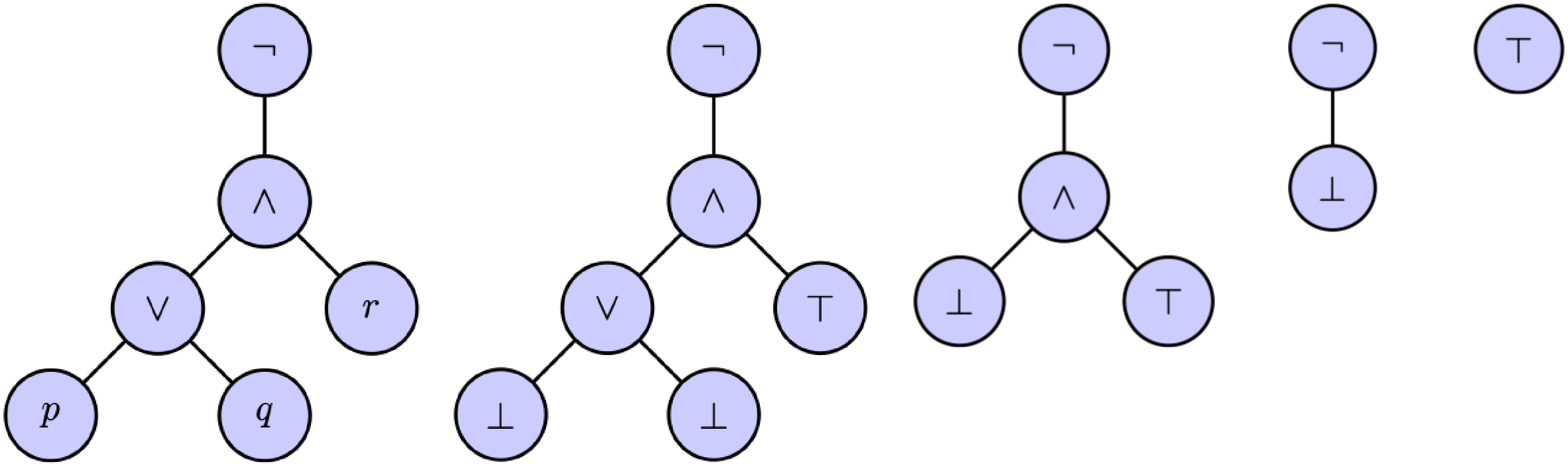
- L'arbre syntaxique ou arbre de syntaxe abstraite permet aussi d'évaluer une expression étant donnée la valuation des variables
  - Exemple : $p = \bot$, $q = \bot$ et $r = \top$



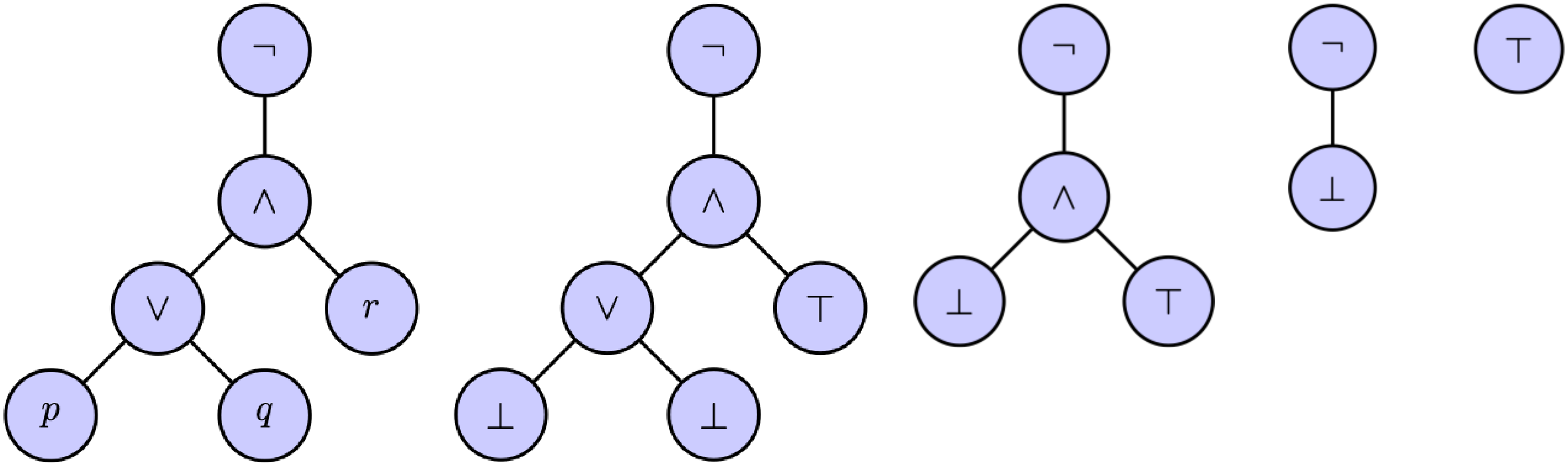
- L'arbre syntaxique ou arbre de syntaxe abstraite permet aussi d'évaluer une expression étant donnée la valuation des variables
  - Exemple : $p = \bot$, $q = \bot$ et $r = \top$



- L'arbre syntaxique ou arbre de syntaxe abstraite permet aussi d'évaluer une expression étant donnée la valuation des variables
  - Exemple : $p = \bot$, $q = \bot$ et $r = \top$



- L'arbre syntaxique ou arbre de syntaxe abstraite permet aussi d'évaluer une expression étant donnée la valuation des variables
  - Exemple : $p = \bot$, $q = \bot$ et $r = \top$



  - Pour un arbre il existe plusieurs stratégies d'évaluation
    - Chaque langage possède la sienne

- Il est fréquent de présenter les valeurs d'une expression sous la forme d'une table où les entrées sont les valuations des variables de l'expression et les sorties la valuation de l'expression
  - Exemple : $E = ((p \Rightarrow \neg q) \vee \neg(q \Leftrightarrow r)) \wedge (p \oplus r)$

- Il est fréquent de présenter les valeurs d'une expression sous la forme d'une table où les entrées sont les valuations des variables de l'expression et les sorties la valuation de l'expression
  - Exemple : $E = ((p \Rightarrow \neg q) \vee \neg(q \Leftrightarrow r)) \wedge (p \oplus r)$

p	q	r	$\neg q$	$p \Rightarrow \neg q$	$q \Leftrightarrow r$	$\neg(q \Leftrightarrow r)$	$(p \Rightarrow \neg q) \vee \neg(q \Leftrightarrow r)$	$p \oplus r$	E
$\bot$	$\bot$	$\bot$	$\top$	$\top$	$\top$	$\bot$	$\top$	$\bot$	$\bot$
$\bot$	$\bot$	$\top$	$\top$	$\top$	$\bot$	$\top$	$\top$	$\top$	$\top$
$\bot$	$\top$	$\bot$	$\bot$	$\top$	$\bot$	$\top$	$\top$	$\bot$	$\bot$
$\bot$	$\top$	$\top$	$\bot$	$\top$	$\top$	$\bot$	$\top$	$\top$	$\top$
$\top$	$\bot$	$\bot$	$\top$	$\top$	$\top$	$\bot$	$\top$	$\top$	$\top$
$\top$	$\bot$	$\top$	$\top$	$\top$	$\bot$	$\top$	$\top$	$\bot$	$\bot$
$\top$	$\top$	$\bot$	$\bot$	$\bot$	$\bot$	$\top$	$\top$	$\top$	$\top$
$\top$	$\top$	$\top$	$\bot$	$\bot$	$\top$	$\bot$	$\bot$	$\bot$	$\bot$

- Quelques propriétés des opérateurs logiques

- Quelques propriétés des opérateurs logiques
  - La négation est involutive
    - $\neg\neg p \Leftrightarrow p$

p	$\neg p$	$\neg\neg p$
$\bot$	$\top$	$\bot$
$\top$	$\bot$	$\top$

- Quelques propriétés des opérateurs logiques
  - La négation est involutive
    - $\neg\neg p \Leftrightarrow p$
  - Les connecteurs $\wedge$ et $\vee$ sont **idempotents**
    - $p \wedge p \Leftrightarrow p$
    - $p \vee p \Leftrightarrow p$

p	p	$p \wedge p$	$p \vee p$
$\perp$	$\perp$	$\perp$	$\perp$
$\top$	$\top$	$\top$	$\top$

- Quelques propriétés des opérateurs logiques
  - La négation est involutive
    - $\neg\neg p \Leftrightarrow p$
  - Les connecteurs $\wedge$ et $\vee$ sont **idempotents**
    - $p \wedge p \Leftrightarrow p$
    - $p \vee p \Leftrightarrow p$
  - Simplifications fondamentales
    - $p \vee \bot \Leftrightarrow p$, $p \vee \top \Leftrightarrow \top$, $p \vee \neg p \Leftrightarrow \top$
    - $p \wedge \bot \Leftrightarrow \bot$, $p \wedge \top \Leftrightarrow p$, $p \wedge \neg p \Leftrightarrow \bot$

p	$\neg p$	$\bot$	$\top$	$p \vee \bot$	$p \vee \top$	$p \vee \neg p$	$p \wedge \bot$	$p \wedge \top$	$p \wedge \neg p$
$\bot$	$\top$	$\bot$	$\top$	$\bot$	$\top$	$\top$	$\bot$	$\bot$	$\bot$
$\top$	$\bot$	$\bot$	$\top$	$\top$	$\top$	$\top$	$\bot$	$\top$	$\bot$

- Quelques propriétés des opérateurs logiques
  - Commutativité, associativité
    - $p \vee q \Leftrightarrow q \vee p$ et $p \wedge q \Leftrightarrow q \wedge p$
    - $(p \wedge q) \wedge r \Leftrightarrow p \wedge (q \wedge r)$

- Quelques propriétés des opérateurs logiques
  - Commutativité, associativité
    - $p \vee q \Leftrightarrow q \vee p$ et $p \wedge q \Leftrightarrow q \wedge p$
    - $(p \wedge q) \wedge r \Leftrightarrow p \wedge (q \wedge r)$
  - Distributivité
    - $p \vee (q \wedge r) \Leftrightarrow (p \vee q) \wedge (p \vee r)$

p	q	r	$q \wedge r$	$p \vee (q \wedge r)$	$p \vee q$	$p \vee r$	$(p \vee q) \wedge (p \vee r)$
$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$
$\bot$	$\bot$	$\top$	$\bot$	$\bot$	$\bot$	$\top$	$\bot$
$\bot$	$\top$	$\bot$	$\bot$	$\bot$	$\top$	$\bot$	$\bot$
$\bot$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$
$\top$	$\bot$	$\bot$	$\bot$	$\top$	$\top$	$\top$	$\top$
$\top$	$\bot$	$\top$	$\bot$	$\top$	$\top$	$\top$	$\top$
$\top$	$\top$	$\bot$	$\bot$	$\top$	$\top$	$\top$	$\top$
$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$	$\top$

- Quelques propriétés des opérateurs logiques
  - Commutativité, associativité
    - $p \vee q \Leftrightarrow q \vee p$ et $p \wedge q \Leftrightarrow q \wedge p$
    - $(p \wedge q) \wedge r \Leftrightarrow p \wedge (q \wedge r)$
  - Distributivité
    - $p \vee (q \wedge r) \Leftrightarrow (p \vee q) \wedge (p \vee r)$
    - $p \wedge (q \vee r) \Leftrightarrow (p \wedge q) \vee (p \wedge r)$

- Quelques propriétés des opérateurs logiques
  - Commutativité, associativité
    - $p \lor q \Leftrightarrow q \lor p$ et $p \land q \Leftrightarrow q \land p$
    - $(p \land q) \land r \Leftrightarrow p \land (q \land r)$
  - Distributivité
    - $p \lor (q \land r) \Leftrightarrow (p \lor q) \land (p \lor r)$
    - $p \land (q \lor r) \Leftrightarrow (p \land q) \lor (p \land r)$
  - Absorption
    - $(p \lor q) \land p \Leftrightarrow p$
    - $(p \land q) \lor p \Leftrightarrow p$

p	q	$p \lor q$	$(p \lor q) \land p$	$p \land q$	$(p \land q) \lor p$
$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$
$\bot$	$\top$	$\top$	$\bot$	$\bot$	$\bot$
$\top$	$\bot$	$\top$	$\top$	$\bot$	$\top$
$\top$	$\top$	$\top$	$\top$	$\top$	$\top$

- Quelques propriétés des opérateurs logiques
  - De Morgan
    - $\neg(p \vee q) \Leftrightarrow \neg p \wedge \neg q$
    - $\neg(p \wedge q) \Leftrightarrow \neg p \vee \neg q$

p	q	$p \vee q$	$\neg(p \vee q)$	$\neg p$	$\neg q$	$\neg p \wedge \neg q$	$p \wedge q$	$\neg(p \wedge q)$	$\neg p \vee \neg q$
$\perp$	$\perp$	$\perp$	$\top$	$\top$	$\top$	$\top$	$\perp$	$\top$	$\top$
$\perp$	$\top$	$\top$	$\perp$	$\top$	$\perp$	$\perp$	$\perp$	$\top$	$\top$
$\top$	$\perp$	$\top$	$\perp$	$\perp$	$\top$	$\perp$	$\perp$	$\top$	$\top$
$\top$	$\top$	$\top$	$\perp$	$\perp$	$\perp$	$\perp$	$\top$	$\perp$	$\perp$

- Simplification
  - $p \vee \neg(p \wedge q)$

- Simplification
  - $p \vee \neg(p \wedge q)$
  - $p \vee \neg p \vee \neg q$ (De Morgan)

- Simplification
  - $p \vee \neg(p \wedge q)$
  - $p \vee \neg p \vee \neg q$ (De Morgan)
  - $(p \vee \neg p) \vee \neg q$ (associativité)

- Simplification
  - $p \vee \neg(p \wedge q)$
  - $p \vee \neg p \vee \neg q$ (De Morgan)
  - $(p \vee \neg p) \vee \neg q$ (associativité)
  - $\top \vee \neg q$ (fondamentale)

- Simplification
  - $p \lor \neg(p \land q)$
  - $p \lor \neg p \lor \neg q$ (De Morgan)
  - $(p \lor \neg p) \lor \neg q$ (associativité)
  - $\top \lor \neg q$ (fondamentale)
  - $\neg q \lor \top$ (commutativité)

- Simplification
  - $p \vee \neg(p \wedge q)$
  - $p \vee \neg p \vee \neg q$ (De Morgan)
  - $(p \vee \neg p) \vee \neg q$ (associativité)
  - $\top \vee \neg q$ (fondamentale)
  - $\neg q \vee \top$ (commutativité)
  - $\top$ (fondamentale)

- Simplification
  - $\neg(p \land q) \land (\neg p \lor q) \land (q \lor \neg q)$

- Simplification
  - $\neg(p \wedge q) \wedge (\neg p \vee q) \wedge (q \vee \neg q)$
  - $\neg(p \wedge q) \wedge (\neg p \vee q) \wedge \top$ (fondamentale)

- Simplification
  - $\neg(p \wedge q) \wedge (\neg p \vee q) \wedge (q \vee \neg q)$
  - $\neg(p \wedge q) \wedge (\neg p \vee q) \wedge \top$ (fondamentale)
  - $\neg(p \wedge q) \wedge (\neg p \vee q)$ (fondamentale)

- Simplification
  - $\neg(p \wedge q) \wedge (\neg p \vee q) \wedge (q \vee \neg q)$
  - $\neg(p \wedge q) \wedge (\neg p \vee q) \wedge \top$ (fondamentale)
  - $\neg(p \wedge q) \wedge (\neg p \vee q)$ (fondamentale)
  - $(\neg p \vee \neg q) \wedge (\neg p \vee q)$ (De Morgan)

- Simplification
  - $\neg(p \land q) \land (\neg p \lor q) \land (q \lor \neg q)$
  - $\neg(p \land q) \land (\neg p \lor q) \land \top$ (fondamentale)
  - $\neg(p \land q) \land (\neg p \lor q)$ (fondamentale)
  - $(\neg p \lor \neg q) \land (\neg p \lor q)$ (De Morgan)
  - $(\neg p \land (\neg p \lor q)) \lor (\neg q \land (\neg p \lor q)$ (distributivité)

- Simplification
  - $\neg(p \wedge q) \wedge (\neg p \vee q) \wedge (q \vee \neg q)$
  - $\neg(p \wedge q) \wedge (\neg p \vee q) \wedge \top$ (fondamentale)
  - $\neg(p \wedge q) \wedge (\neg p \vee q)$ (fondamentale)
  - $(\neg p \vee \neg q) \wedge (\neg p \vee q)$ (De Morgan)
  - $(\neg p \wedge (\neg p \vee q)) \vee (\neg q \wedge (\neg p \vee q)$ (distributivité)
  - $((\neg p \wedge \neg p) \vee (\neg p \wedge q)) \vee ((\neg q \wedge \neg p) \vee (\neg q \wedge q))$ (distributivité)

- Simplification
  - $\neg(p \wedge q) \wedge (\neg p \vee q) \wedge (q \vee \neg q)$
  - $\neg(p \wedge q) \wedge (\neg p \vee q) \wedge \top$ (fondamentale)
  - $\neg(p \wedge q) \wedge (\neg p \vee q)$ (fondamentale)
  - $(\neg p \vee \neg q) \wedge (\neg p \vee q)$ (De Morgan)
  - $(\neg p \wedge (\neg p \vee q)) \vee (\neg q \wedge (\neg p \vee q))$ (distributivité)
  - $((\neg p \wedge \neg p) \vee (\neg p \wedge q)) \vee ((\neg q \wedge \neg p) \vee (\neg q \wedge q))$ (distributivité)
  - $(\neg p \vee (\neg p \wedge q)) \vee ((\neg q \wedge \neg p) \vee (\neg q \wedge q))$ (idempotent)

- Simplification
  - $\neg(p \wedge q) \wedge (\neg p \vee q) \wedge (q \vee \neg q)$
  - $\neg(p \wedge q) \wedge (\neg p \vee q) \wedge \top$ (fondamentale)
  - $\neg(p \wedge q) \wedge (\neg p \vee q)$ (fondamentale)
  - $(\neg p \vee \neg q) \wedge (\neg p \vee q)$ (De Morgan)
  - $(\neg p \wedge (\neg p \vee q)) \vee (\neg q \wedge (\neg p \vee q)$ (distributivité)
  - $((\neg p \wedge \neg p) \vee (\neg p \wedge q)) \vee ((\neg q \wedge \neg p) \vee (\neg q \wedge q))$ (distributivité)
  - $(\neg p \vee (\neg p \wedge q)) \vee ((\neg q \wedge \neg p) \vee (\neg q \wedge q))$ (idempotent)
  - $(\neg p \vee (\neg p \wedge q)) \vee ((\neg q \wedge \neg p) \vee \perp)$ (fondamentale)

- Simplification
  - $\neg(p \wedge q) \wedge (\neg p \vee q) \wedge (q \vee \neg q)$
  - $\neg(p \wedge q) \wedge (\neg p \vee q) \wedge \top$ (fondamentale)
  - $\neg(p \wedge q) \wedge (\neg p \vee q)$ (fondamentale)
  - $(\neg p \vee \neg q) \wedge (\neg p \vee q)$ (De Morgan)
  - $(\neg p \wedge (\neg p \vee q)) \vee (\neg q \wedge (\neg p \vee q)$ (distributivité)
  - $((\neg p \wedge \neg p) \vee (\neg p \wedge q)) \vee ((\neg q \wedge \neg p) \vee (\neg q \wedge q))$ (distributivité)
  - $(\neg p \vee (\neg p \wedge q)) \vee ((\neg q \wedge \neg p) \vee (\neg q \wedge q))$ (idempotent)
  - $(\neg p \vee (\neg p \wedge q)) \vee ((\neg q \wedge \neg p) \vee \perp)$ (fondamentale)
  - $(\neg p \vee (\neg p \wedge q)) \vee (\neg q \wedge \neg p)$ (fondamentale)

- Simplification
  - $\neg(p \wedge q) \wedge (\neg p \vee q) \wedge (q \vee \neg q)$
  - $\neg(p \wedge q) \wedge (\neg p \vee q) \wedge \top$ (fondamentale)
  - $\neg(p \wedge q) \wedge (\neg p \vee q)$ (fondamentale)
  - $(\neg p \vee \neg q) \wedge (\neg p \vee q)$ (De Morgan)
  - $(\neg p \wedge (\neg p \vee q)) \vee (\neg q \wedge (\neg p \vee q)$ (distributivité)
  - $((\neg p \wedge \neg p) \vee (\neg p \wedge q)) \vee ((\neg q \wedge \neg p) \vee (\neg q \wedge q))$ (distributivité)
  - $(\neg p \vee (\neg p \wedge q)) \vee ((\neg q \wedge \neg p) \vee (\neg q \wedge q))$ (idempotent)
  - $(\neg p \vee (\neg p \wedge q)) \vee ((\neg q \wedge \neg p) \vee \bot)$ (fondamentale)
  - $(\neg p \vee (\neg p \wedge q)) \vee (\neg q \wedge \neg p)$ (fondamentale)
  - $\neg p \vee (\neg q \wedge \neg p)$ (absorption)

- Simplification
  - $\neg(p \wedge q) \wedge (\neg p \vee q) \wedge (q \vee \neg q)$
  - $\neg(p \wedge q) \wedge (\neg p \vee q) \wedge \top$ (fondamentale)
  - $\neg(p \wedge q) \wedge (\neg p \vee q)$ (fondamentale)
  - $(\neg p \vee \neg q) \wedge (\neg p \vee q)$ (De Morgan)
  - $(\neg p \wedge (\neg p \vee q)) \vee (\neg q \wedge (\neg p \vee q)$ (distributivité)
  - $((\neg p \wedge \neg p) \vee (\neg p \wedge q)) \vee ((\neg q \wedge \neg p) \vee (\neg q \wedge q))$ (distributivité)
  - $(\neg p \vee (\neg p \wedge q)) \vee ((\neg q \wedge \neg p) \vee (\neg q \wedge q))$ (idempotent)
  - $(\neg p \vee (\neg p \wedge q)) \vee ((\neg q \wedge \neg p) \vee \perp)$ (fondamentale)
  - $(\neg p \vee (\neg p \wedge q)) \vee (\neg q \wedge \neg p)$ (fondamentale)
  - $\neg p \vee (\neg q \wedge \neg p)$ (absorption)
  - $\neg p$ (absorption)