Logique mathématique classique

Cours 1 : Introduction

Université Mohamed Seddik Ben Yahia -Jijel-

2ième année Licence informatique

F. BOUDJERIDA

Plan du cours

- Pourquoi un cours de logique ?
- Quelques exemples
- Quelques repères historiques
- Déroulement du cours

Introduction

- Qu’est ce qu’une logique ?
- Objectif de la logique mathématique
- Pourquoi un cours de logique ?
- Langages

Qu’est ce qu’une logique ?

- Le mot logique vient du grecque « logos » qui signifie parole discours.
- Logique est la science qui étude les règles qui doivent respecter tout raisonnement valide et qui permet de distinguer un raisonnement valide, d’un raisonnement qui ne l’est pas donc c’est l’art de raisonner correctement
- Logique mathématique elle s’intéresse à l’organisation et à la cohérence du discours mathématique c-à-d aux notions de validité

Objectif de la logique mathématique

Les objectifs de la logique sont de

– Traiter formellement les notions de vérité et fausseté
– Formaliser et justifier le raisonnement logique intuitif, la déduction logique
– Permettre le raisonnement (humain ou automatique) dans des cas non intuitifs, complexes

Pourquoi un cours de logique ?

La logique apparaît en de nombreuses circonstances en systèmes d'information.

- **Spécification formelle des systèmes**
- **Vérification du logiciel**
- **Logique des données**
- **Systèmes basés sur la connaissance**
- **Logique de la connaissance** : définition formelle des concepts, définition de théories (le temps, l'espace, les matériaux, la physique, la gestion, …), modélisation logique

Pourquoi un cours de logique ?

En outre, les fondements théoriques de l'informatique et de la logique formelle sont fortement liés. Notre compréhension profonde de la logique.

- La logique en informatique est plus généralement une base de l'intelligence artificielle.
- Le but est de rendre les machines capable d'appliquer le raisonnement.
- Pour décrire les problèmes de ce domaines ; les nombres et les fonctions mathématiques ne suffisent pas. Nous avons besoin d'utiliser un langage.

Langages

Il y a deux types de langages :

1. Langage naturel
2. Langage formel

Langage naturel

- Est le langage que nous utilisons dans la vie ; de tous les jours
- Ce langage a deux inconvénients majeurs qu'on utilise en mathématique ;
  - la complexité des phrases qui rend les choses plus compliqué, il faux plusieurs lignes incompréhensible pour dire quelque chose qui peut se résumer une simple équation
  - les ambigüités du langage qui peuvent conduire à des erreurs

Langage naturel

Alors l'idée est d'utiliser un langage symbolique permet de simplifier de choses en mathématique.

Langage formel

Lorsque l’on définit on doit définir deux choses qui caractérisent ce langage :

- syntaxe
- sémantique

Langage formel

→ syntaxe

- **Un alphabet** : un ensemble de symboles
- **Une grammaire** : un ensemble de règles qui définit quels mots appartiennent au langage formel

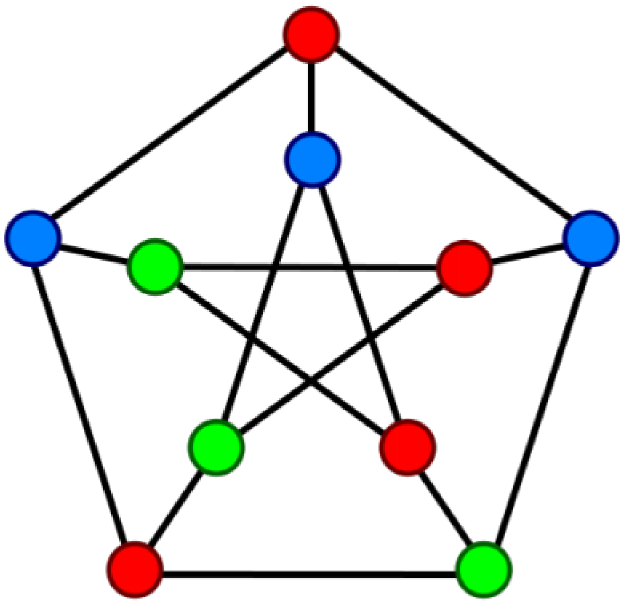
Mot : est une suite ordonnée de symboles, ces symboles appartenant à l'alphabet

- Un langage formel est un ensemble des mots de longueur finie par un alphabet et une grammaire

Exp : langage propositionnelle, langage des prédicats, …..

Quelques exemples

Problème coloration de graphe



Graphe (de Petersen)

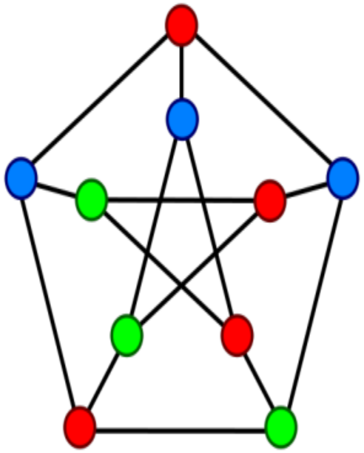
Coloration de graphe

attribuer une couleur à chacun de ses sommets t. q.

deux sommets reliés par une arête soient de couleurs différentes

Quelques exemples

Problème coloration de graphe



Graphe (de Petersen)

représentation

10 sommets : $S_1, \cdots S_{10}$, 3 couleurs : *Rouge*, *Bleu*, *Vert*

$S_{i,R}$ Vrai si S_i est de couleur Rouge, $1 \leq i \leq 10$

$S_{i,B}$ Vrai si S_i est de couleur Bleu, $1 \leq i \leq 10$

$S_{i,V}$ Vrai si S_i est de couleur Vert, $1 \leq i \leq 10$

représentation

chaque sommet a au moins une couleur :

$S_{i,R} \vee S_{i,B} \vee S_{i,V}$

chaque sommet a au plus une couleur :

$S_{i,R} \rightarrow (\neg S_{i,B} \wedge \neg S_{i,V})$

$S_{i,B} \rightarrow (\neg S_{i,R} \wedge \neg S_{i,V})$

$S_{i,V} \rightarrow (\neg S_{i,R} \wedge \neg S_{i,B})$

Quelques exemples

représentation

chaque sommet a au moins une couleur :

$S_{i,R} \vee S_{i,B} \vee S_{i,V}$

chaque sommet a au plus une couleur :

$S_{i,R} \rightarrow (\neg S_{i,B} \wedge \neg S_{i,V})$

$S_{i,B} \rightarrow (\neg S_{i,R} \wedge \neg S_{i,V})$

$S_{i,V} \rightarrow (\neg S_{i,R} \wedge \neg S_{i,B})$

représentation

les sommets voisins ont des couleurs différentes

pour tout couple de sommets voisins (S_i, S_j) :

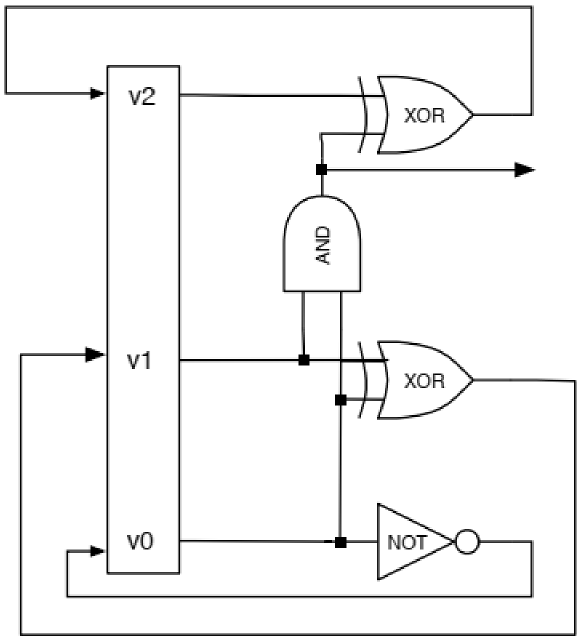
$S_{i,R} \rightarrow \neg S_{j,R}$

$S_{i,B} \rightarrow \neg S_{j,B}$

$S_{i,V} \rightarrow \neg S_{j,V}$

Quelques exemples

Représentation de circuits (source : N. Baudru)



Compteur modulo 8

Circuit synchrone

- toutes les transitions s'effectuent en même temps
- l'état prochain de chaque registre peut donc être calculé en fonction de l'état courant des registres.

Quelques exemples

Modelisation d'un circuit synchrone (source : N. Baudru)

- L'ensemble des variables est constitué des sorties des registres plus les entrées primaires : $V = \{v_0, v_1, v_2\}$. Soit $V' = \{v'_0, v'_1, v'_2\}$ la copie de V.
- L'état initial est $\mathcal{S}_0 \equiv \neg v_0 \wedge \neg v_1 \wedge \neg v_2$
- Pour les transitions, on décrit quelles prochaines valeurs peuvent prendre les registres en fonction de leur valeur courante :
  - $\mathcal{R}_0 \equiv v'_0 \iff \neg v_0$
  - $\mathcal{R}_1 \equiv v'_1 \iff v_0 \oplus v_1$
  - $\mathcal{R}_2 \equiv v'_2 \iff (v_0 \wedge v_1) \oplus v_2$

  Puisque tous ces changements se produisent en même temps (circuit synchrone) il faut combiner ces contraintes pour obtenir la relation finale : $\mathcal{R} \equiv \mathcal{R}_0 \wedge \mathcal{R}_1 \wedge \mathcal{R}_2$

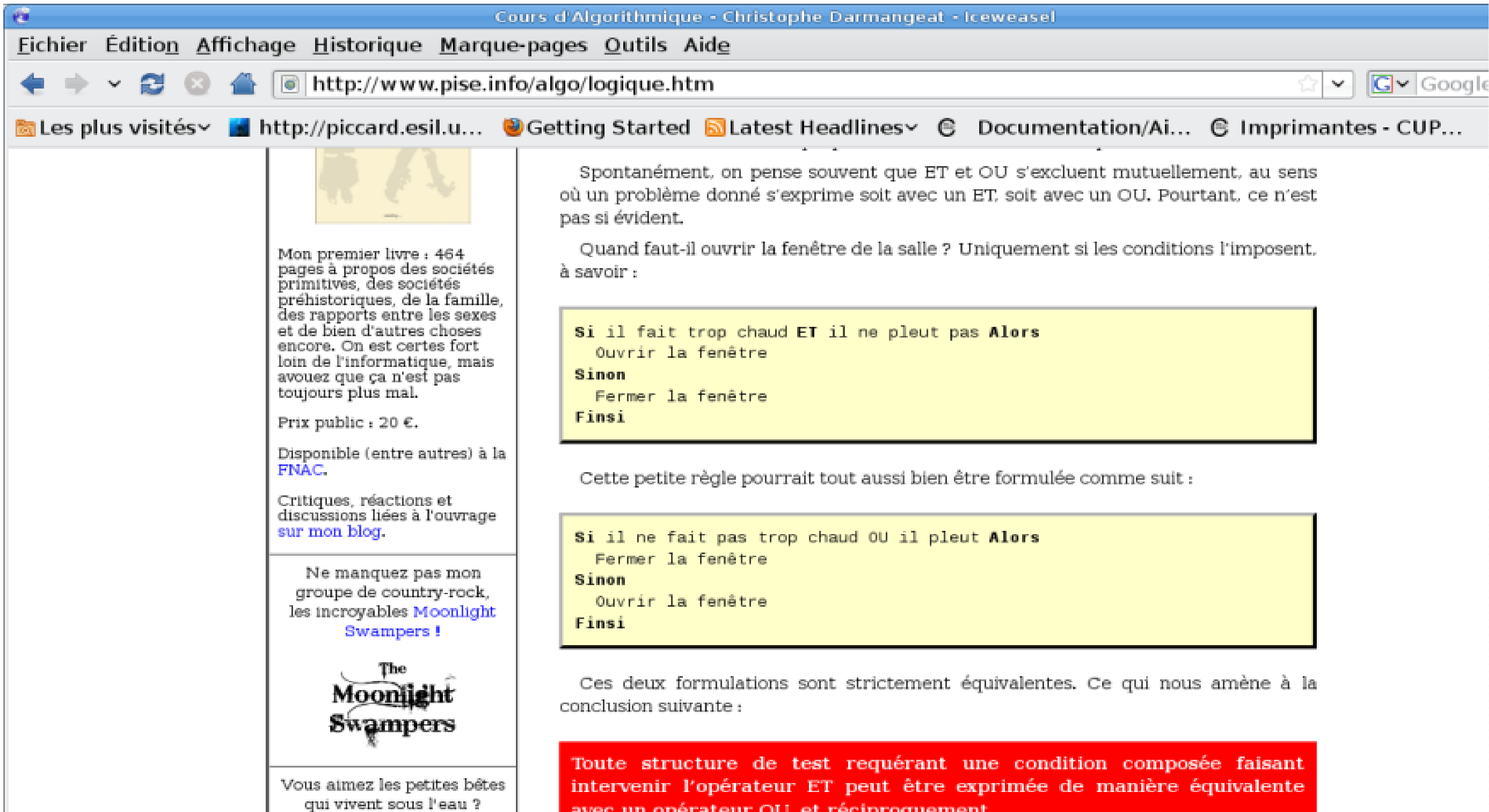
Quelques exemples

Algorithmique

Ecriture d'algorithmes

- booléens
- expressions booléennes
- conjonctions : ··· **et** ···, disjonctions : ··· **ou** ···
- négation d'expressions booléennes complexes
- expressions conditionnelles : **si** ··· **alors** ··· **sinon** ···

Logique et algorithmique (source : C. Darmangeat)



Cours d'Algorithmique - Christophe Darmangeat - Iceweasel

Fichier Édition Affichage Historique Marque-pages Outils Aide

http://www.pise.info/algo/logique.htm

Les plus visités http://piccard.esil.u... Getting Started Latest Headlines Documentation/Ai... Imprimantes - CUP...

Mon premier livre : 464 pages à propos des sociétés primitives, des sociétés préhistoriques, de la famille, des rapports entre les sexes et de bien d'autres choses encore. On est certes fort loin de l'informatique, mais avouez que ça n'est pas toujours plus mal.

Prix public : 20 €.

Disponible (entre autres) à la FNAC.

Critiques, réactions et discussions liées à l'ouvrage sur mon blog.

Ne manquez pas mon groupe de country-rock, les incroyables Moonlight Swampers !

The Moonlight Swampers

Vous aimez les petites bêtes qui vivent sous l'eau ?

Spontanément, on pense souvent que ET et OU s'excluent mutuellement, au sens où un problème donné s'exprime soit avec un ET, soit avec un OU. Pourtant, ce n'est pas si évident.

Quand faut-il ouvrir la fenêtre de la salle ? Uniquement si les conditions l'imposent, à savoir :

```
Si il fait trop chaud ET il ne pleut pas Alors
  Ouvrir la fenêtre
Sinon
  Fermer la fenêtre
Finsi
```

Cette petite règle pourrait tout aussi bien être formulée comme suit :

```
Si il ne fait pas trop chaud OU il pleut Alors
  Fermer la fenêtre
Sinon
  Ouvrir la fenêtre
Finsi
```

Ces deux formulations sont strictement équivalentes. Ce qui nous amène à la conclusion suivante :

Toute structure de test requérant une condition composée faisant intervenir l'opérateur ET peut être exprimée de manière équivalente avec un opérateur OU, et réciproquement.

Complexité algorithmique

Informatique théorique

Le problème SAT

Etant donnée une formule booléenne mise sous forme normale conjonctive, existe t-il une affectation de valeurs (0 ou 1) des variables booléennes rendant la formule vraie ?

SAT est un problème NP-Complet de référence [Cook 1971]

Complexité algorithmique

Réduction au problème SAT

Pour montrer qu'un problème est NP-complet on le transforme en problème SAT

problème OLNE

$0 \leq x_1 \leq 1$

$0 \leq x_2 \leq 1$

$x_1 + x_2 \geq 1$

$x_1 + (1 - x_2) \geq 1$

$(1 - x_1) + (1 - x_2) \geq 1$

représentation

x_i tq $0 \leq x_i \leq 1$: variable v_i

pour chaque équation :

$x_1 + \cdots + x_k + (1 - x_{k+1}) + \cdots + (1 - x_n) \geq 1$

associer la formule :

$v_1 \vee \cdots \vee \neg v_{k+1} \vee \cdots \vee \neg v_n$

Réduction au problème SAT

problème OLNE

$$0 \leq x_1 \leq 1$$
$$0 \leq x_2 \leq 1$$
$$x_1 + x_2 \geq 1$$
$$x_1 + (1 - x_2) \geq 1$$
$$(1 - x_1) + (1 - x_2) \geq 1$$

représentation

variables : v_1, v_2

$F = (v_1 \vee v_2) \wedge (v_1 \vee \neg v_2) \wedge (\neg v_1 \vee \neg v_2)$

F est satisfaisable

solution problème OLNE :
$x_i = 0$ si $v_i = Faux$,
$x_i = 1$ si $v_i = Vrai$

Quelques repères historiques

une brève retrospective

AC *IV* ième Antiquité	**Fondements : les propositions** Aristote, les Stoïciens
XIII – *XIV* ième Moyen-âge	**Logique scholastique : dialectique** R. Lulle, G. d'Occam
XV – *XIX* ième	apports de Leibniz, Euler
XIX ième	**Mathématisation de la logique** G. Boole, de Morgan, Frege
*XX*ième	**Logique contemporaine** B. Russell, K. Goëdel

Logiques contemporaines

Logiques classiques

- logique propositionnelle
- logique des prédicats

Logiques non classiques

- logiques **modales**
  - connecteurs supplémentaires : *modalités*
    représenter le temps, l'espace, les permissions, les obligations, les interdictions $\cdots$
- loqiques **multivaluées**
  - autres valeurs de vérité que *Vrai* et *Faux* :
    Indéterminé, Probable, Plausible, $\cdots$

Quelques repères historiques

- **Logique propositionnelle :**

Le calcul des propositions est un système formel dans lequel les formules représentent des propositions qui peuvent être formées en combinant les propositions atomiques et en utilisant les connecteurs logiques, et dans lequel un système de règles de démonstration formelle établit certains «théorèmes».

- **Calcul des prédicats:**

Un calcul des prédicats est un système formel, qui peut être soit la logique du premier ordre, soit la logique du second ordre, soit la logique d'ordre supérieur, soit la logique infinitaire. Il exprime par la quantification un large échantillon de propositions du langage naturel.

Déroulement du cours

- Chapitre1: Introduction
- Chapitre2: Logique propositionnelle
- Chapitre3: Raisonnement en logique propositionnelle
- Chapitre4: Logique des prédicats
- Chapitre5: Raisonnement en logique des prédicats