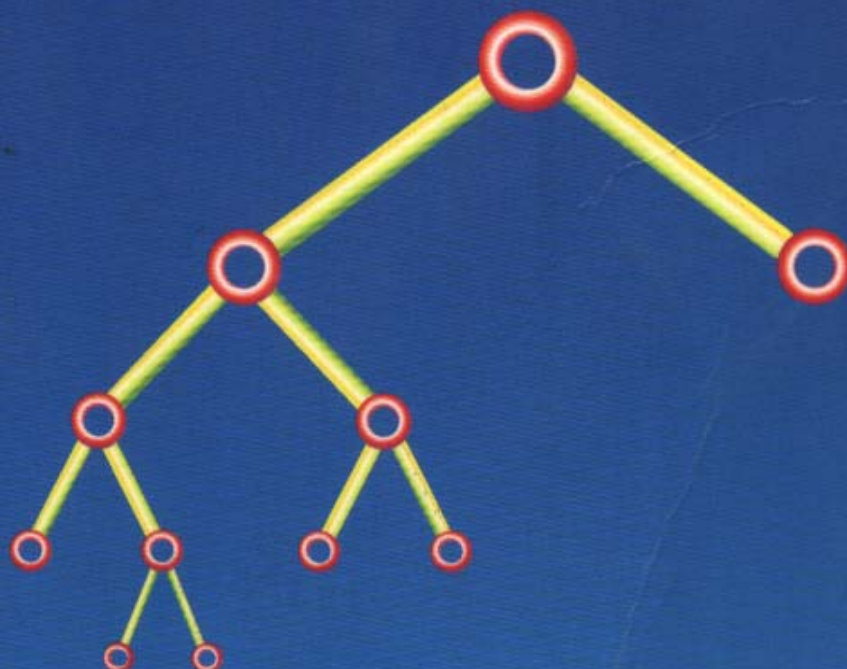


Logique et mathématiques

pour l'informatique et l'I.A.

109 exercices corrigés

C. JACQUEMIN



MASSON 

TABLE DES MATIÈRES

(see Contents page vii)

1. Structure d'ordre et treillis	1
1.1. Structure d'ordre	1
1.1.1. Relation d'ordre	1
1.1.2. Graphe d'ordre	3
1.2. Treillis	4
1.2.1. Définition et exemples	4
1.2.2. Différents types de treillis	5
1.3. Algèbre de Boole, calcul booléen	6
1.3.1. Algèbre de Boole	6
1.3.2. Isomorphisme entre un treillis booléen et $P(E)$	8
1.4. Fonction booléenne	10
1.4.1. Introduction	10
1.4.2. Forme canonique des fonctions booléennes	13
1.4.3. Formes canoniques et réalisation de circuits	15
1.4.4. Simplification des fonctions booléennes	17
1.4.5. Méthodes de recherche des monômes premiers	18
1.5. Exercices	21
1.5.1. Exercices sur les structures d'ordre	21
1.5.2. Exercices sur les treillis	26
1.5.3. Exercices sur les algèbres de Boole et le calcul booléen	32
1.5.4. Exercices sur les fonctions booléennes	33
2. Logique	41
2.1. Systèmes formels	41
2.1.1. Définition des systèmes formels	42
2.1.2. Interprétation	43
2.2. Calcul propositionnel	44
2.2.1. Langage propositionnel (syntaxe)	44
2.2.2. Calcul des propositions (sémantique)	45
2.2.3. Système formel de la logique des propositions	48
2.2.4. Méthode de résolution de Robinson	49
2.3. Calcul des prédicats du premier ordre	51
2.3.1. Langage des prédicats du premier ordre	51
2.3.2. Variables libres ou liées	53
2.3.3. Interprétation du calcul des prédicats	54
2.3.4. Système formel de la logique des prédicats	56
2.4. Principe de résolution et Prolog	57
2.4.1. Principe de résolution en calcul des prédicats	58
2.4.2. Le langage Prolog	61
2.5. Exercices	64
2.5.1. Exercices sur les systèmes formels	64
2.5.2. Exercices sur le calcul propositionnel	67
2.5.3. Exercices sur le calcul des prédicats	74
2.5.4. Exercices sur le principe de résolution et Prolog	78
3. Relations n-aires	87
3.1. Relation n-aire	87

3.1.1.	Définition d'une relation <i>n</i> -aire	87
3.1.2.	Relation <i>n</i> -aire non orientée	89
3.2.	Opérations sur les relations <i>n</i> -aires	91
3.2.1.	Définitions des opérations	91
3.2.2.	Projection	94
3.2.3.	Sélection	95
3.3.	Dépendance fonctionnelle et clé	96
3.3.1.	Dépendance fonctionnelle	96
3.3.2.	Clés	97
3.4.	Exercices	99
4.	Théorie des langages	109
4.1.	Structure de monoïde	109
4.1.1.	Définition	110
4.1.2.	Trois propriétés des monoïdes	110
4.1.3.	Sous-monoïde	111
4.2.	Langages et grammaires	112
4.2.1.	Algorithme, machine et langage	112
4.2.2.	Syntaxe d'un langage	113
4.2.3.	Sémantique d'un langage	113
4.3.	Grammaires	114
4.3.1.	Définition	114
4.3.2.	Notion de dérivation	115
4.3.3.	Grammaires régulières et hors contexte	117
4.3.4.	Propriétés sur les grammaires	119
4.4.	Automates	121
4.4.1.	Automates à états finis	121
4.4.2.	Automates à états finis déterministes	123
4.4.3.	Expressions régulières	125
4.4.4.	Automates à pile	128
4.5.	Introduction à la compilation	132
4.6.	Exercices	142
4.6.1.	Exercices sur la structure de monoïde	142
4.6.2.	Exercices sur les grammaires	147
4.6.3.	Exercices sur les automates et les expressions régulières	157
5.	Graphes	171
5.1.	Listes, piles, files et arbres	171
5.1.1.	Listes	172
5.1.2.	Piles	175
5.1.3.	Files	177
5.1.4.	Arbres	178
5.2.	Graphes	184
5.2.1.	Graphes orientés	184
5.2.2.	Graphes non orientés	190
5.3.	Algorithmes sur les graphes	193
5.3.1.	Problème du plus court chemin	193
5.3.2.	Algorithme de Moore-Dijkstra	194
5.3.3.	Applications en ordonnancement	197
5.3.4.	Cycles et chaînes eulériens	201
5.4.	Exercices	203
5.4.1.	Exercices sur les listes, les piles, et les files	203
5.4.2.	Exercices sur les arbres	208
5.4.3.	Exercices sur les graphes	214
5.4.4.	Exercices sur les algorithmes sur les graphes	223
	Bibliographie	233
	Index	235

CONTENTS

1. Ordered sets and lattices	1
1.1. Ordered sets	1
1.1.1. Order relations	1
1.1.2. Graph of an order relation	3
1.2. Lattices	4
1.2.1. Definition and example	4
1.2.2. Some types of lattices	5
1.3. Boolean algebras, Boolean calculus	6
1.3.1. Boolean algebras	6
1.3.2. Isomorphism between a Boolean lattice and the power set $P(E)$	8
1.4. Boolean functions	10
1.4.1. Introduction	10
1.4.2. Canonical factors	13
1.4.3. Canonical factors and circuit design	15
1.4.4. Simplifying Boolean functions	17
1.4.5. Prime factors of a Boolean function	18
1.5. Exercises	21
1.5.1. Exercises on ordered sets	21
1.5.2. Exercises on lattices	26
1.5.3. Exercises on Boolean algebras and Boolean calculus	32
1.5.4. Exercises on Boolean functions	33
2. Logic	41
2.1. Formal systems	41
2.1.1. Definition of a formal system	42
2.1.2. Interpretation	43
2.2. Statement logic	44
2.2.1. Syntax	44
2.2.2. Semantics	45
2.2.3. The formal system of statement logic	48
2.2.4. Robinson's resolution principle	49
2.3. Predicate logic	51
2.3.1. Syntax	51
2.3.2. Bound and free variables	53
2.3.3. Semantics	54
2.3.4. The formal system of predicate logic	56
2.4. Robinson's resolution principle and Prolog	57
2.4.1. Robinson's resolution principle for predicate logic	58
2.4.2. Prolog language	61
2.5. Exercises	64
2.5.1. Exercises on formal systems	64
2.5.2. Exercises on statement logic	67
2.5.3. Exercises on predicate logic	74
2.5.4. Exercises on Robinson's resolution principle and Prolog	78
3. n-ary relations	87
3.1. n -ary relations	87
3.1.1. Definition of an n -ary relation	87
3.1.2. Non oriented n -ary relations	89

3.2.	Operations on n -ary relations	91
3.2.1.	Definition of some operations	91
3.2.2.	Projection.....	94
3.2.3.	Selection.....	95
3.3.	Functional dependencies and keys.....	96
3.3.1.	Functional dependencies	96
3.3.2.	Keys.....	97
3.4.	Exercises.....	99
4.	Languages.....	109
4.1.	Monoid structures.....	109
4.1.1.	Definition.....	110
4.1.2.	Three properties of monoids.....	110
4.1.3.	Submonoids.....	111
4.2.	Languages and grammars.....	112
4.2.1.	Algorithm, machine and language	112
4.2.2.	Syntax of a language	113
4.2.3.	Semantics of a language	113
4.3.	Grammars	114
4.3.1.	Definition.....	114
4.3.2.	Notion of derivation	115
4.3.3.	Regular and context free grammars	117
4.3.4.	Some properties of grammars.....	119
4.4.	Automata.....	121
4.4.1.	Finite automata	121
4.4.2.	Deterministic finite automata	123
4.4.3.	Regular languages	125
4.4.4.	Pushdown automata.....	128
4.5.	Introduction to compilers	132
4.6.	Exercises.....	142
4.6.1.	Exercises on monoid structures.....	142
4.6.2.	Exercises on grammars.....	147
4.6.3.	Exercises on automata and regular languages	157
5.	Graphs	171
5.1.	Lists, stacks, queues and trees	171
5.1.1.	Lists.....	172
5.1.2.	Stacks.....	175
5.1.3.	Queues.....	177
5.1.4.	Trees	178
5.2.	Graphs.....	184
5.2.1.	Directed graphs	184
5.2.2.	Non-directed graphs	190
5.3.	Algorithms.....	193
5.3.1.	Shortest path problem.....	193
5.3.2.	Moore-Dijkstra's algorithm.....	194
5.3.3.	Application to time-scheduling problems.....	197
5.3.4.	Euler circuit and chain.....	201
5.4.	Exercises.....	203
5.4.1.	Exercises on lists, stacks, queues and trees	203
5.4.2.	Exercises on trees.....	208
5.4.3.	Exercises on graphs	214
5.4.4.	Exercises on algorithms	223
References		233
Index		235

MEMO-GUIDES

La conception et la réalisation de systèmes informatiques requièrent des compétences en mathématiques discrètes et en logique. Ce *Mémo-guide* présente les définitions, les principes et les méthodes mathématiques de base qui s'appliquent à la résolution de problèmes posés par le traitement automatique de l'information. Ces méthodes sont utilisées en algorithmique et en programmation, en conception des systèmes d'information, en intelligence artificielle et dans l'étude de l'architecture des ordinateurs.

Cinq chapitres, indépendants les uns des autres, réunissent les principaux outils mathématiques : le calcul booléen et les structures d'ordre et de treillis, la logique propositionnelle, la logique des prédicats et le modèle mathématique du langage Prolog, les relations n -aires, les langages et analyseurs formels, les algorithmes sur les graphes.

Cet aide-mémoire est assorti de nombreux exemples et exercices corrigés, de difficulté progressive.

Les étudiants en informatique des IUT, ainsi que des licences, maîtrises, écoles d'ingénieurs... trouveront dans cet ouvrage une synthèse pédagogique utile à toute tâche informatique de formalisation et d'automatisation.

*Professeur agrégé de mathématiques, **Christian JACQUEMIN** est maître de conférences en informatique, à l'IUT de Nantes.*

