

COURS ET EXERCICES D'INFORMATIQUE

Classes préparatoires, 1^{er} et 2nd cycles universitaires

Ouvrage collectif — Coordination Luc Albert





Cours et exercices d'informatique

جامعة أمحمد بوقرة - بومرداس
Université M'hamed Bougara - Boumerdes

المكتبة الجامعية

رقم: 004(075.8)/A1B

5-22

جامعة أمحمد بوقرة - بومرداس
Université M'hamed Bougara - Boumerdes

المكتبة الجامعية

رقم الجرد: 0059878

Table des matières

Préface

xiii

Le malentendu informatique	xiii
Le programme d'informatique des classes préparatoires	xiv
Le contenu de ce livre	xvi

I Présentation de CAML

1

1	Caml est un langage interactif	2
2	Caml est un langage (fortement) typé	4
3	Caml est un langage fonctionnel	8
3.1	Les fonctions Caml	8
3.2	Les variables	10
3.3	Le filtrage	12
4	C'est aussi un langage impératif	16
4.1	Le type unit	16
4.2	La conditionnelle	17
4.3	Les références	18
4.4	Les boucles	19
4.4.1	La boucle inconditionnelle	19
4.4.2	La boucle conditionnelle	20
4.5	Délimiteurs	22
4.6	La programmation impérative est-elle indispensable?	24
5	Compléments	24
5.1	Variables de type	24
5.1.1	Polymorphisme	25
5.1.2	Variables de type faibles	27
5.2	Fonctionnelles, opérateurs, curryfication	28
5.3	Filtrage et égalité	31

5.4	Type défini par l'utilisateur	34
5.4.1	Type somme et type produit (enregistrement)	34
5.4.2	Type mutable	36
5.4.3	Type avec argument	36
5.4.4	Type polymorphe	37
5.4.5	Type récursif	38
5.4.6	Types mutuellement récursifs	40
5.5	Entrées-sorties, flots, compilation	41
5.5.1	Entrées-sorties	41
5.5.2	Flots	43
5.5.3	Compilation	46
5.6	Valeurs exceptionnelles	47
5.6.1	Gestion des exceptions	47
5.6.2	Utilisation des exceptions en programmation	51
5.7	Débogage	53
5.8	Graphisme	58
6	Conclusion	62
	Exercices	63
	Correction	68

II Éléments de programmation 77

1	Quelques conseils pour une « bonne » programmation...	77
2	Récursivité	79
2.1	Introduction	79
2.2	Problèmes de terminaison	82
2.2.1	Si peu de mathématiques...	83
2.2.2	Quelques exemples	85
2.2.3	Indécidabilité de la terminaison	87
2.3	Preuves sur les programmes récursifs	88
2.4	Programmer c'est prouver	91
2.5	Récursivité croisée	94
3	Programmes impératifs, éléments de preuve	95
4	Itération contre récursivité	97
	Exercices	104
	Correction	114

III Structures de données 129

1	Labyrinthe et graphe	129
1.1	Implémentation des graphes	130
1.2	Le parcours en profondeur d'abord	134
1.3	Le fil d'Ariane...	139
2	Type de données abstrait	141
2.1	Définitions	141
2.2	Induction structurelle	143
3	Structures linéaires	149

3.1	Les structures de données « liste »	149
3.1.1	Liste dynamique	149
3.1.2	La fonctionnelle <code>list_it</code>	150
3.1.3	Liste indexée	152
3.1.4	Recherche dans une liste	153
3.2	Les piles	154
3.2.1	Spécifications	156
3.2.2	Implémentations	157
3.2.3	Exemples d'utilisation	159
3.3	Les files	160
3.3.1	Spécifications	161
3.3.2	Implémentations	161
3.3.3	Exemples d'utilisation	164
4	Arbres	165
4.1	Définitions	165
4.1.1	Une approche intuitive	165
4.1.2	Définition formelle des arbres binaires	167
4.1.3	Le type de données abstrait arbre binaire	168
4.2	Algorithmes de base	169
4.3	Parcours d'arbres binaires	170
4.3.1	Parcours en profondeur	171
4.3.2	Traitements préfixe, infixe, postfixe	172
4.3.3	Parcours en largeur	172
4.4	Combinatoire des arbres binaires	174
4.5	Arbres binaires de recherche	176
4.5.1	Définition	176
4.5.2	Recherche	176
4.5.3	Insertion	177
4.5.4	Suppression	177
4.5.5	Profondeur moyenne	179
	Exercices	181
	Correction	189

IV Complexité des algorithmes

207

1	Position du problème	207
2	Choix d'un système de mesure	208
3	Complexité moyenne et complexité dans le pire des cas	208
4	Une étude de cas : la recherche linéaire	209
5	Outils mathématiques	210
6	Les principales classes de complexité	211
7	Attention aux coûts occultes !	212
8	Résolution des récurrences linéaires	213
9	Résolution des récurrences <i>diviser pour régner</i>	214
9.1	Terminologie et exemples	214
9.2	Étude du cas où n est une puissance de 2	215
9.3	Étude du cas général	215

10	Complexité du tri rapide	216
10.1	Complexité du tri rapide dans le pire des cas	216
10.2	Complexité en moyenne du tri rapide	216
10.3	Conclusion sur les tris	217
	Exercices	218
	Correction	223

V Automates finis 231

1	Quelques définitions	231
2	Qu'est-ce qu'un automate?	234
2.1	Définition et représentation des automates	236
2.2	Langage accepté par un automate	237
2.3	Automates complets	240
2.4	Automates émondés	242
2.5	Calcul explicite de l'émondé d'un automate	243
2.6	Commentaires	246
3	Automates déterministes	247
3.1	Définition et premières propriétés	248
3.2	Déterminisation d'un automate	250
3.3	Commentaires	254
4	Automates avec ε -transitions	255
5	Automates et motifs	258
6	Propriétés de fermeture	263
6.1	Opérations ensemblistes	263
6.2	Concaténation et itération	265
6.3	Quotients	269
6.4	Morphismes	272
7	Critères de reconnaissabilité	274
8	Langages et expressions rationnels	280
8.1	Langages rationnels	280
8.2	Expressions rationnelles	281
9	Théorème de Kleene	284
10	Minimisation	291
10.1	Simulation	292
10.2	Équivalence de Nérade	294
10.2.1	Définition	294
10.2.2	Calcul pratique	296
10.3	Automate des résiduels	300
10.4	Existence et unicité de l'automate minimal	301
10.5	Cas des automates non-déterministes	302
	Correction	307

VI Logique propositionnelle

331

1	Syntaxe	332
1.1	Structure arborescente d'une proposition	332
1.2	Notation parenthésée	334
1.3	Autres notations linéaires	337
2	Sémantique	337
2.1	Évaluation	338
2.2	Tables de vérité	340
	Note culturelle	341
3	Tautologies	342
3.1	Tautologies	342
3.1.1	Définitions	342
3.1.2	Exemples fondamentaux de tautologies	342
3.2	Obtention de tautologies	344
3.2.1	Substitution	344
3.2.2	Principe de dualité	344
3.3	Méthodes usuelles de démonstration	345
3.3.1	Analyse de cas	345
3.3.2	Contraposition	345
3.3.3	Raisonnement par l'absurde	346
3.3.4	Raisonnement par équivalence à <i>Vrai</i>	346
3.4	Démonstration automatique	347
3.4.1	Déduction	347
3.4.2	Résolution	348
4	Formes normales	351
4.1	Fonctions logiques	351
4.2	Nouvelles notations	352
4.3	Formes conjonctives et disjonctives d'une proposition	352
5	Circuits logiques	355
5.1	Portes logiques fondamentales	356
5.2	Circuits combinatoires	357
5.3	Tableaux de Karnaugh	358
5.4	Exemples de circuits combinatoires	361
5.4.1	Codeurs. Décodeurs. Transcodeurs	361
5.4.2	Multiplexeurs	364
5.4.3	Circuits arithmétiques	366
5.4.4	Additionneur « diviser pour régner »	368
5.5	Notions sur les circuits séquentiels	375
	Exercices	378
	Correction	392

<i>A Programme de l'option informatique en MPSI</i>	413
<i>B Programme de l'option informatique en MP</i>	417
<i>C Notations</i>	423
<i>Bibliographie</i>	425
<i>Index</i>	429

[...] il n'y a plus que deux manières de se servir d'un Livre comme il faut. La première est la même dont plusieurs gens usent à l'égard des grands seigneurs : ils apprennent par cœur leurs titres, et ensuite ils se vantent d'en être les amis intimes. La seconde, qui est la mieux choisie, et la plus profonde, consiste à s'attacher à la Table des Matières, par laquelle un livre est dirigé, comme un vaisseau par le gouvernail.

Swift — *Le conte du tonneau*