

K.L. Clark

F.G. McCabe

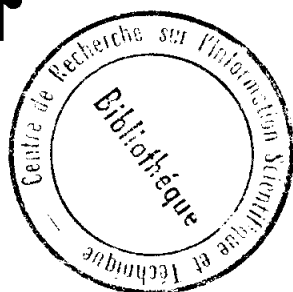
micro-PROLOG:
programmer
en logique

BIBLIOTHEQUE DU CERIST

Editions EYROLLES

K.L. Clark F.G. McCabe

micro-PROLOG: programmer en logique



TRADUIT DE L'ANGLAIS
par **Michel ROUSSEAU**


EYROLLES

61, boulevard Saint-Germain – 75005 Paris
1985

Préface

Ce livre constitue un manuel d'auto-initiation à la programmation logique utilisant le langage micro-PROLOG (version 3.1 ou suivante) et est destiné aux personnes qui ne sont pas familiarisées avec la programmation de ce langage. Les concepts sous-tendant la programmation logique ainsi que les éléments leur correspondant en micro-PROLOG y sont introduits pas à pas par le biais d'une série de programmes exemples.

Des exercices, dont les solutions sont données en fin d'ouvrage, viennent appuyer les exemples. Pour bien faire il faudrait suivre exemples et exercices sur votre ordinateur, mais, rassurez-vous, ce n'est pas indispensable.

micro-PROLOG est couramment disponible sur les machines équipées d'un Z80 sous CP/M80 et sur les ordinateurs dotés d'un 8088/86 sous MSDOS. On peut également se le procurer pour programmer sur Sinclair SPECTRUM, sur BBC ainsi que sur les autres ordinateurs fonctionnant sous UNIX.

Puisque micro-PROLOG fait partie de la famille des langages de programmation logique (PROLOG est en effet l'abrégé de PROgrammation LOGique), chacun de ces "idiomes" dérivant de la version du langage originellement implantée par Alain COLMAURER en 1972/73 alors qu'il était professeur du centre de recherches sur l'Intelligence Artificielle à l'Université d'Aix-Marseille, ce livre peut également vous servir d'introduction à la programmation logique de n'importe quelle autre version de PROLOG. Les différences entre micro-PROLOG et les autres versions de ce langage

résident au plus dans la syntaxe des programmes ainsi que dans les formes autorisées d'interrogation.

Structure de l'ouvrage

Ce livre est divisé en quatre parties. La première partie nous présente les concepts fondamentaux de la programmation logique tout en mettant l'accent sur l'emploi de cette logique, et plus particulièrement de micro-PROLOG, en tant que langage de description et d'interrogation des données. La seconde partie s'occupe de concepts plus avancés ainsi que de leurs pendants en micro-PROLOG. L'accent est ici mis sur l'utilisation des processus de liste en logique. La troisième partie aborde la syntaxe standard de micro-PROLOG, syntaxe utilisée pour compiler les programmes des parties précédentes. Cette partie décrit également les éléments du langage qui permettent l'implantation d'autres systèmes de programmation logique au sommet de micro-PROLOG. Enfin la quatrième partie héberge quatre chapitres traitant d'applications de micro-PROLOG. Ils montrent comment utiliser micro-PROLOG pour jouer à des jeux, pour résoudre des problèmes ainsi que dans le cadre des systèmes experts et dans celui de l'analyse des phases critiques.

Pour plus de détails sur le contenu des chapitres de chaque section reportez-vous au Chapitre 0, Chapitre d'introduction qui précède la première partie. Ce Chapitre vous donnera déjà un avant-goût du style de programmation qui caractérise la programmation logique.

Note-système - utilisation conjointe de l'ouvrage et d'un ordinateur - si vous disposez d'un appareil supportant micro-PROLOG, vous désirerez sans doute suivre les exemples et les exercices de cet ouvrage sur votre ordinateur. Pour vous le permettre, nous avons inclus une série de *notes-système* (comme celle-ci) portant sur l'emploi effectif de micro-PROLOG sur un ordinateur. D'ordinaire, les notes-système font plutôt référence à des activités de programmation non-logique comme l'interruption de l'exécution d'un programme.

Si vous utilisez un ordinateur vous devrez consulter de temps à autre le Manuel de référence concernant votre implantation; ceci est dû au fait que cet ouvrage ne décrit que les caractéristiques du langage disponibles sur toute implantation et ne peut bien entendu décrire chaque caractéristique propre à une machine donnée, comme par exemple l'utilisation de l'éditeur de ligne.

Table des matières

PREFACE	VII
INTRODUCTION	1
0.1 POURQUOI PROGRAMMER EN micro-PROLOG	3
Langages descriptifs contre langages impératifs	5
Définitions multi-usages	6
Bases de données	7
Programmation réglementée et modélisée	7
Traitement de liste	8
Fonctions impératives	8
0.2 DESCRIPTION DU CONTENU DE L'OUVRAGE	9
Première Partie : concepts de base	9
Deuxième Partie : Programmation logique au moyen de micro-PROLOG	10
Troisième Partie : micro-PROLOG mode d'emploi	11
Quatrième Partie : Applications de micro-PROLOG	12
 <u>PREMIERE PARTIE : CONCEPTS ELEMENTAIRES</u>	
CHAPITRE I : FAITS ET QUESTIONS	13
1.1 DEVELOPPEMENT D'UNE BASE DE DONNEES CONCERNANT DES FAITS	15
Ajouter des faits	15
<i>Différents types de relations</i>	18
Ajoutons encore quelques faits	20
<i>L'instruction accept</i>	21
Lister et sauvegarder un programme	22
<i>Edition par ajout ou effacement d'assertions</i>	24
Edition au moyen de l'éditeur de lignes	26
Résumé des instructions de développement des programmes	26
<i>Exercices 1-1</i>	29
1.2 QUESTIONS	31
Confirmation	31

<i>Les variables dans les questions</i>	32
Extraction de données	33
Questions conjonctives (ou jointes)	34
Résumé des instructions d'interrogation	36
<i>Trouver les noms de vos relations</i>	38
<i>Relations prédéfinies et modules</i>	38
<i>Exercices 1-2</i>	39
1.3 RELATIONS ARITHMETIQUES	40
Addition et soustraction au moyen de la relation SUM	41
<i>Utilisation de la relation SUM</i>	41
<i>Restrictions à l'usage de SUM</i>	42
<i>Syntaxe des nombres</i>	42
Conversion et tests des types numériques	43
<i>Utilisation de INT</i>	44
<i>Restrictions dans l'utilisation de INT</i>	44
Multiplication et division à l'aide de TIMES	45
<i>Utilisations de la relation TIMES</i>	45
<i>Restrictions à l'usage des questions TIMES</i>	46
Utilisation de la relation LESS pour vérifier l'ordre	46
<i>Utilisations de LESS</i>	46
<i>Exercices 1-3</i>	47
1.4 EVALUATION DES QUESTIONS	47
Evaluation des questions ne comportant qu'une condition	48
Question is à un modèle d'assertion	49
Evaluation des questions which n'ayant qu'une condition	51
Evaluation des questions conjonctives which	53
Evaluation des questions conjonctives is	57
<i>Exercices 1-4</i>	59
1.5 DES QUESTIONS EFFICACES	61
CHAPITRE II : REGLES	63
2.1 TRANSFORMER LES QUESTIONS EN REGLES	65
Utilisation de plusieurs règles	67
<i>Des variables dans les règles</i>	69
Assertions conditionnelles	69
Lecture logique d'une règle conditionnelle	70
Lecture de contrôle de : S si C	71
<i>Exercices 2-1</i>	71
Des règles peuvent utiliser des relations définies par d'autres règles	72

	<i>Le programme, arrivé à ce point</i>	73
	<i>Exercices 2-2</i>	74
	<i>En savoir plus sur les modèles de réponse</i>	74
2.2	COMMENT SONT EVALUEES LES QUESTIONS COMPORTANT DES REGLES	76
	Chaînage arrière	77
	Règles	78
	<i>Résumé de la méthode d'évaluation</i>	79
	Evaluation type	79
2.3	DESCRIPTIONS RECURSIVES DES RELATIONS	83
	Définition séparée de fonctions inverses	85
	<i>Exercices 2-3</i>	88
	Description récursive des relations arithmétiques	89
	<i>Définition récursive d'un intervalle d'entiers</i>	91
	Définition de la factorielle inverse	93
	<i>Exercices 2-4</i>	94
	CHAPITRE III : LISTES	97
3.1	LISTE EN TERMES D'INDIVIDUS	99
	<i>Exercices 3-1</i>	100
3.2	ACCEDER AUX MEMBRES D'UNE LISTE DE LONGUEUR DETERMINEE	101
	<i>Exercices 3-2</i>	103
	Listes de listes	104
	<i>Termes</i>	105
3.3	ACCEDER AUX MEMBRES D'UNE LISTE DE LONGUEUR INCONNUE	105
	Têtes et queues	106
	<i>Exercices 3-3</i>	108
	Appartient-à	108
	<i>Exercices 3-4</i>	111
	<i>Autres utilisations de "appartient-à"</i>	112
	<i>Lecture de contrôle des règles "appartient-à"</i>	112
3.4	LA LONGUEUR D'UNE LISTE	113
	<i>Lectures de contrôle</i>	115
	<i>Conclusion</i>	118
	<i>Exercices 3-5</i>	120

Unification et relation EQ	120
Construction d'une chaîne de descendants	122
<i>Exercices 3-6</i>	123
3.5 ENSEMBLES DE REPONSES SOUS FORME DE LISTES	123
La condition complexe isall	124
<i>Exercices 3-7</i>	125

DEUXIEME PARTIE : PROGRAMMATION LOGIQUE EN micro-PROLOG

CHAPITRE IV : CONDITIONS COMPLEXES DANS LES QUESTIONS ET DANS LES REGLES	129
4.1 CONDITIONS D'INVERSION LOGIQUE	131
Syntaxe des conditions d'inversion logique	132
Limitations à l'utilisation de not	133
<i>Lecture de contrôle</i>	133
Egalités inverses	135
Vérification et non génération de règles	135
not avec "appartient-à"	136
<i>Exercices 4-1</i>	137
4.2 LA CONDITION isall	138
Syntaxe des conditions isall	139
<i>Restrictions d'utilisation</i>	139
<i>Utilisation à des fins de génération</i>	139
<i>Utilisation à des fins de vérification</i>	139
<i>Lecture de contrôle</i>	141
Utilisation de isall et de "appartient-à"	142
<i>Exercice 4-2</i>	142
<i>Générer et vérifier</i>	143
4.3 LA CONDITION forall	145
Syntaxe des conditions forall	145
<i>Lecture logique</i>	145
<i>Restrictions d'utilisation</i>	145
<i>Lecture de contrôle</i>	146
Double négation équivalente	146
Exemples d'utilisation de forall	147
<i>Exercices 4-3</i>	148
4.4 LA CONDITION OR	149
Syntaxe des conditions or	150
Des règles uniques pour des relations de listes	150

	<i>Exercices 4-4</i>	151
4.5	CONDITIONS D'EXPRESSION	151
	Conditions d'égalité	152
	Syntaxe des expressions	153
	<i>Syntaxe des expressions arithmétiques</i>	154
	<i>Exemples d'expressions arithmétiques</i>	155
	Expressions d'appel de fonction	155
	<i>Syntaxe des appels de fonction</i>	156
	Avertissement	157
	<i>Expressions imbriquées</i>	157
	Conditions d'expression	158
	<i>Syntaxe des conditions expressives</i>	158
	Forme compilée des conditions d'expression	159
	Relations arithmétiques spéciales	160
	Evaluation des conditions d'expression	161
	La question d'expression #	161
	<i>Exercices 4-5</i>	162
4.6	INTERROGATION DE L'UTILISATEUR AU MOYEN DE IS-TOLD	163
	Forme générale des conditions is-told	167
	<i>Exercices 4-6</i>	168
	Développement de programmes de a à z	169
4.7	CONDITIONS DE DOCUMENTATION	170
	La condition commentaire /*	170
	La relation commentaire vars	171
	Avertissement	172
	Assertions documentaires	172
	CHAPITRE V : TRAITEMENT DE LISTE	175
5.1	LA RELATION D'ADJONCTION	177
	Utilisation de l'adjonction pour éclater une liste	180
	<i>Exercices 5-1</i>	181
5.2	DES REGLES QUI UTILISENT L'ADJONCTION	182
	<i>Exercices 5-2</i>	186
5.3	DEFINITION RECURSIVE DE LA RELATION DE TRI	187
	Tri d'interclassement	188
	Tri rapide	191
	<i>Exercices 5-3</i>	192

5.4	FONCTIONS DE LISTE	193
	Utilisation de / dans des expressions	193
CHAPITRE VI: INTRODUCTION A L'ANALYSE SYNTAXIQUE		195
6.1	ASSERTIONS D'ANALYSE SYNTAXIQUE EXPRIMEES SOUS FORME DE LISTES DE MOTS	197
	Trouver des phrases	201
	<i>Exercices 6-1</i>	202
6.2	UN AUTRE PROGRAMME D'ANALYSE SYNTAXIQUE	203
	<i>Exercices 6.2</i>	206
	Trouver des phrases	206
6.3	UTILISATION GENERALE DES COUPLES DIFFERENTIELS	208
	<i>Exercices 6.3</i>	209
CHAPITRE VII : DE QUELQUES CONSIDERATIONS PRAGMATIQUES		211
7.1	LIMITER UNE CONDITION A UNE SEULE SOLUTION	214
	Forme générale d'une condition à solution unique	215
	<i>Conditions-test de solution unique</i>	215
	<i>Des conditions à solution unique dans les règles</i>	216
	Versions test des relations	216
	<i>Exercices 7-1</i>	217
7.2	CONTROLE DU CHAINAGE ARRIERE AU MOYEN D'UNE CONDITION /	217
	/ dans les règles	217
	<i>Utilisation type de / dans les règles</i>	218
	<i>Utilisation de / pour définir des relations de test</i>	220
	Interroger l'utilisateur par défaut au sujet des relations test	221
7.3	PILE DE QUESTIONS ET SAUVEGARDE DE L'ESPACE MEMOIRE	221
7.4	DEFINITIONS RECURSIVES DE FIN DE LISTE	224
	Forme générale d'une définition de fin de liste	224

Autres définitions récursives de fin de liste	225
Changement en forme récursive de fin de liste	225
Généralisation des relations	226
Définitions récursives semi-terminales	228
Exercices 7-2	229
7.5 UTILISATION DE MODULES	229
Construire un module	232
Exercice 7-3	234
CHAPITRE VIII : PROGRAMMATION META-LOGIQUE	235
8.1 NOMS DE RELATION ET LISTES D'ARGUMENTS EN GUISE DE VARIABLES	237
Variable en tant que liste d'arguments	238
Nom de relation en tant que variable	238
La méta-condition true-of	238
Exemples	239
Exercices 8-1	239
Programmes généralisés	239
Exercices 8-2	241
Une relation générale de confrontation de listes	242
8.2 META-PROGRAMMES VERIFIANT LES CONDITIONS D'UTILISATION	242
Sélectionner une définition pour un usage déterminé	243
Exercices 8-3	244
8.3 DES PROGRAMMES QUI EN MANIPULENT D'AUTRES	245
Exercice 8-4	249
Conserver les réponses à une question sous forme de faits	249
Utilisation de delete	250
Une base de données en guise de mémoire brouillon	250
Exercices 8-5	253
Des variables dans les assertions	253
8.4 DES RELATIONS UNAIRES POUR INSTRUCTIONS	254
Les relations impératives de lecture/écriture	256
La primitive R	257
Autre exemple d'utilisation	258
Les primitives P et PP	259

<i>Impression des variables</i>	261
<i>Lire ou écrire dans des fichiers</i>	262
Une définition plus élaborée de "is-told"	262
Utilisation des primitives de lecture/écriture	
afin de définir de nouvelles instructions	263
Le programme superviseur intégré	264
<i>Définition de relations à deux arguments</i>	265
<i>Exercices 8-6</i>	266

★ TROISIEME PARTIE : micro-PROLOG MODE D'EMPLOI

CHAPITRE IX : LA SYNTAXE STANDARD DE micro-PROLOG	267
9.1 ATOMES ET CLAUSES	270
<i>Exemples d'atomes représentant des assertions simples</i>	271
<i>Exemples d'assertions sous forme de clauses</i>	272
<i>Exemples de conditions complexes sous forme d'atomes</i>	272
9.2 PROGRAMMER EN SYNTAXE STANDARD	273
Entrer des clauses	273
<i>Lister des clauses</i>	275
<i>Exercice 9-1</i>	277
<i>La primitive d'interrogation ?</i>	278
9.3 ANALYSE DE LA TRANSFORMATION D'ASSERTIONS EN CLAUSES	280
La relation Conv-d'A.S	280
La relation Analyse-de-Cond-Conj	281
La relation A-d-A	281
9.4 DES META-VARIABLES DANS LES PROGRAMMES EN SYNTAXE STANDARD	283
Méta-variable remplaçant le prédicat d'un atome	284
<i>Exercices 9-2</i>	286
Méta-variable en tant qu'atome	287
Méta-variable comme partie restante d'une clause	288
<i>Exercice 9-3</i>	289
Méta-variable en tant que reste de la liste d'arguments	289
<i>Exemple d'utilisation</i>	289
9.5 CL , LA PRIMITIVE D'ACCES AUX CLAUSES	292

Définition de nouvelles relations interrogatives	294
Exercice 9-4	295

QUATRIEME PARTIE : APPLICATIONS DE micro-PROLOG

CHAPITRE X : UN PROGRAMME D'ANALYSE DES CHEMINS CRITIQUES	297
Introduction	299
10.1 POSER LE PROBLEME	300
10.2 DEFINITION DES CONCEPTS EN micro-PROLOG	303
10.3 UTILISATION D'EN-TETES	310
10.4 LE PROGRAMME D'ANALYSE DES CHEMINS CRITIQUES	315
CHAPITRE XI : micro-PROLOG ET LES SYSTEMES EXPERTS	319
11.1 INTRODUCTION	321
11.2 LE SYSTEME EXPERT MYCIN	322
Règle 1	322
11.3 REPRESENTATION DES CONNAISSANCES EN micro-PROLOG	323
<i>Noyau pour un système expert</i>	323
Représentation de la connaissance sous forme de règle	324
Règle 2	324
Règle 3	324
Règle 4	324
Règle 5	325
Règle 6	326
11.4 UN NOYAU INTERACTIF SIMPLE	327
Généralisation du programme "est-rapporté"	327
Contrôler l'interaction	329
11.5 UN NOYAU PLUS SOPHISTIQUE	331
Le noyau devrait savoir quoi demander	331
Questions interactives "IS" et "WHICH"	332
11.6 EXPLICATIONS "pouquoi"	333
Qu'est-ce qu'une explication "pourquoi"?	333

Génération d'un traçage de "pourquoi"	334
Afficher le traçage de "pourquoi"	337
11.7 EXPLICATIONS "comment"	338
Qu'est-ce qu'une explication "comment" ?	338
Implantation d'une explication "comment" simple	340
Améliorer l'explication "comment"	342
11.8 EXPLICATIONS "pourquoi-pas"	343
Implantation d'une explication "pourquoi-pas" simple	343
Amélioration de l'explication "pourquoi-pas"	345
Bibliographie	345
CHAPITRE XII : LA LOGIQUE DES JEUX A DEUX	347
12.1 INTRODUCTION	349
12.2 ARBRES DE JEU, ARBRES FORCES ET PRINCIPE DU MINIMAX	353
Arbres forcés	355
Calcul minimax des valeurs 1, -1	358
12.3 SEUILS ET ELAGAGE ALPHA-BETA	363
Comportement du Calculateur de Mouvement III	367
12.4 REMARQUES POUR CONCLURE	369
✓ CHAPITRE XIII : micro-PROLOG ET LA RESOLUTION DE PROBLEMES	373
Introduction	375
13.1 RESOLUTION DE PROBLEMES PAR RECHERCHE	376
13.2 UNE AUTRE FORMULATION DE LA DECOUVERTE DE CHEMINS	381
Une solution au problème de traversée de la rivière	386
Le problème des réservoirs	387
13.3 AUTRES STRATEGIES DE RECHERCHE	389
Recherche en profondeur	391
Recherche en largeur	392
Stratégies générales de recherche	393
13.4 INCORPORATION DE LA DETECTION DE BOUCLES	393
13.5 RECHERCHE DIRECTE ET EXPERTISE	396

CORRIGE DES EXERCICES	399
INDEX	399
TABLE DES NOTES SYSTEME	431