

INITIATION
A LA
PROGRAMMATION

Claude DELANNOY



EYROLLES

BIBLIOTHEQUE DU CERIST

J.P. Daral

CHEZ LE MEME EDITEUR

Du même auteur

- DELANNOY - *Apprendre à programmer en Fortran* - 200 p. ; 1983.
- DELANNOY - *Apprendre à programmer en Basic* - 272 p. ; 1984.
- DELANNOY - *Les fichiers en Basic sur micro-ordinateur* - 168 p. ; 1984, (coll. Micro-ordinateurs).
- DELANNOY - *Faites vos jeux avec Atmos* - 224 p. ; 1984, (coll. Microplus).
- DELANNOY - *Faites vos jeux avec Oric* - 224 p. ; 1984, (coll. Microplus).

Autres ouvrages

- BONNIN - *Le Cobol A.N.S. avec exercices et corrigés* - 200 p. ; 1984, (coll. Pratique de l'Informatique).
- BOUSSARD et MAHL - *Programmation avancée. Algorithmique et structures de données* - 248 p. ; 1984.
- BRODIE - *Débutez en Forth* - 328 p. ; 1984.
- CANAL - *Parler L.S.E. et apprendre à l'utiliser* - 160 p. ; 1983, (coll. Micro-ordinateurs).
- DAX - *Langage C* - 192 p. ; 1984.
- HERNANDEZ - *Pascal par l'exemple* - 152 p. ; 1983, (coll. Micro-ordinateurs).
- MONTEIL - *Premiers pas en Logo* - 104 p. ; 1984, (coll. Micro-ordinateurs).
- QUEINNEC - *Langage d'un autre type : Lisp* - 200 p. ; 1984, (coll. Micro-ordinateurs).
- SCHOMBERG - *Le Basic universel* - 128 p. ; 1983, (coll. Micro-ordinateurs).
- WEIDENFEL, MATHIEU et PEROLAT - *LOGO* - 160 p. ; 1984.

INITIATION A LA PROGRAMMATION

par

Claude DELANNOY


EYROLLES

61, boulevard Saint-Germain — 75005 Paris

1984

Avant-Propos

La programmation consiste, avant tout, à déterminer la démarche permettant d'obtenir, à l'aide d'un ordinateur, la solution d'un problème donné. C'est ce que l'on nomme généralement « l'analyse » ou encore « la recherche d'algorithme ». La traduction de cette analyse dans un langage de programmation donné n'en constitue que l'aboutissement et la codification suivant des règles précises.

Or, faute de disposer d'un moyen rationnel lui permettant d'exprimer sa démarche, le programmeur en est souvent réduit à « penser » dans le langage qu'il a appris en premier. Dans ces conditions, les contraintes spécifiques du langage, ses particularités techniques, voire sa complexité, masquent nécessairement les aspects fondamentaux de l'analyse.

L'objectif de ce livre est de fournir à tout programmeur (en herbe ou confirmé) les éléments de base intervenant en programmation, quel que soit le langage employé : variable ; type ; instructions d'affectation, de lecture, d'écriture ; structures. En même temps, il en propose un moyen d'expression qui, par rapport à un véritable langage, reste simple et naturel.

Nous avons toujours cherché à rendre concrète chaque notion étudiée, en montrant comment elle se rattachait à la nature même de l'ordinateur. Notre expérience de l'enseignement nous a prouvé, en effet, que cette façon de faire en favorisait grandement l'assimilation, voire dans certains cas l'acceptation. Ceci justifie notamment la présence et l'importance du premier chapitre qui présente le rôle de l'ordinateur, les grandes lignes de son fonctionnement et la manière de l'utiliser. Il dégage les importantes notions de langage, programme, données et résultats.

Les chapitres 2 à 4 sont consacrés à l'étude des trois instructions fondamentales : affectation, lecture et écriture. Les chapitres 5 et 6 présentent les structures de choix et de répétition.

Les chapitres 7 à 9 abordent deux techniques utilisées très fréquemment en programmation : le comptage et l'accumulation.

Le chapitre 10 constitue une approche de l'itération, méthode très puissante qui est la clé de la résolution de nombreux problèmes.

Enfin, les chapitres 11 à 13 étudient les tableaux à un ou à deux indices.

Nous tenons tout particulièrement à attirer l'attention du lecteur sur le fait que la programmation, malgré son caractère fondamental, est une discipline qui ne peut se maîtriser que par la pratique. Certes, ce manuel comporte de nombreux exemples et exercices résolus qui viennent illustrer les différents points abordés. Ce n'est toutefois que par la recherche personnelle des exercices proposés que le lecteur pourra prétendre parvenir à programmer. Nous lui conseillons vivement de ne consulter les solutions fournies en fin de volume qu'à titre de vérification de l'exactitude de sa réflexion.

Une telle démarche devrait permettre au débutant d'aborder l'étude ultérieure d'un ou plusieurs langages de programmation sur des bases saines et solides. Il évitera ainsi l'apprentissage empirique qui a été le lot de beaucoup de ses aînés.

Quant au programmeur confirmé qui a appris son métier sur le tas, il trouvera ici matière à faire le point afin de mieux asseoir ses connaissances.

Table des matières

AVANT-PROPOS.....	VII
I. Ordinateur et programmation	1
1. <i>Le rôle de l'ordinateur</i>	2
1.1. La multiplicité des applications	2
1.2. Le programme : source de diversité	2
1.3. Les données du programme, les résultats	3
1.4. Communication ou archivage ?	3
1.5. Schémas récapitulatifs	4
2. <i>Pour donner une forme à l'information : la notion de codage</i>	5
2.1. L'ordinateur code l'information.....	5
2.2. Il n'est pas le seul.....	6
2.3. Mais.....	7
3. <i>Comment fonctionne l'ordinateur ?.....</i>	8
3.1. A chacun son rôle.....	8
3.2. La mémoire centrale	10
3.3. L'unité centrale	11
3.4. Les périphériques	12
4. <i>Comment parler à l'ordinateur ?.....</i>	14
4.1. Dans son langage ou dans le nôtre?.....	14
4.2. En langage assembleur	15
4.3. En langage évolué	15
4.4. De toutes façons, il faut traduire... ..	16
5. <i>Qu'est-ce que la programmation ?</i>	16
5.1. Plusieurs façons de dire la même chose	17
5.2. Ne pas confondre : connaître un langage et savoir programmer	17
5.3. Analyse et transcription dans un langage	18
5.4. L'objectif de ce manuel	19

II. La notion de variable	21
1. <i>La variable ou comment donner un nom à un emplacement mémoire</i>	21
2. <i>Les noms de variable : un choix important</i>	22
3. <i>Attention aux habitudes de l'algèbre</i>	23
4. <i>Type d'une variable</i>	24
4.1. La notion de type, conséquence du codage binaire	24
4.2. Le type impose ses limitations	25
4.3. Le type limite les opérations	26
4.4. En résumé	27
4.5. Déclaration de type	27
III. L'instruction d'affectation	29
1. <i>Introduction</i>	29
2. <i>Choisissons une notation</i>	30
3. <i>Rôle de l'instruction d'affectation</i>	30
Exercices	32
4. <i>Quelques précautions</i>	32
Exercices	33
5. <i>Les expressions apparaissant dans l'instruction d'affectation</i> ..	35
5.1. Les expressions numériques	35
5.2. Les expressions de type caractère	36
Exercices	37
6. <i>Les variables non définies</i>	37
IV. Pour communiquer avec votre programme : les instructions d'écriture et de lecture	39
1. <i>L'instruction d'écriture pour la communication programme-utilisateur</i>	39
1.1. Son rôle	39
1.2. Présentation des résultats	40
Exercices	42
2. <i>L'instruction de lecture pour la communication utilisateur-programme</i>	42
2.1. Son rôle	42
2.2. Intérêt de l'instruction de lecture	43
2.3. Présentation des données	44
Exercices	46
3. <i>L'écriture de « libellés »</i> . Pour y voir plus clair :	46
3.1. Dans la présentation des résultats	46
Exercices	48
3.2. Dans l'entrée des données sur une console	48
Exercices	49
V. Le choix : première structure fondamentale	51
1. <i>Les structures fondamentales</i>	51

2. <i>La structure de choix : exemples introductifs</i>	53
3. <i>Les conventions d'écriture</i>	54
4. <i>La condition du choix</i>	55
4.1. <i>Les conditions simples</i>	55
Exercices	57
4.2. <i>Les conditions complexes</i>	57
Exercices	58
5. <i>Cas particulier : quand l'une des parties du choix est absente</i>	59
6. <i>Les choix imbriqués</i>	60
VI. La structure : répétition jusqu'à	63
1. <i>Exemple introductif</i>	63
2. <i>Nos conventions d'écriture</i>	64
3. <i>Quand on veut une réponse correcte</i>	65
Exercices	66
4. <i>Pour faire une recherche</i>	66
Exercices	67
5. <i>Faire des choix dans une boucle</i>	68
Exercices	69
VII. Compter	71
1. <i>Le comptage systématique</i>	71
1.1. <i>Le problème</i>	71
1.2. <i>Comment compter ?</i>	72
1.3. <i>Le programme correspondant</i>	73
Exercices	74
2. <i>Le comptage sélectif</i>	74
Exercices	75
3. <i>L'utilisation de plusieurs compteurs</i>	75
Exercices	77
VIII. Une nouvelle structure : la boucle avec compteur	79
1. <i>Exemple : répéter 50 fois les mêmes instructions</i>	79
2. <i>La structure de boucle avec compteur</i>	81
Exercices	84
3. <i>Compter dans une boucle avec compteur</i>	84
Exercices	84
4. <i>Utiliser le compteur dans la boucle</i>	85
Exercices	85
5. <i>Quand le nombre de répétitions est variable</i>	86
Exercices	86
6. <i>Quelques pièges à éviter</i>	87
6.1. <i>Respecter ses engagements</i>	87
6.2. <i>Un tour pour rien ?</i>	88

IX. Accumuler	91
1. <i>L'accumulation systématique</i>	91
Exercices	93
2. <i>L'accumulation sélective</i>	94
Exercices	94
X. L'itération	97
1. <i>Quand nous faisons de l'itération sans le savoir</i>	97
2. <i>Un exemple simple d'itération</i>	98
3. <i>Un autre exemple : calcul de factorielle</i>	101
4. <i>La démarche itérative</i>	104
Exercices	106
5. <i>Application : recherche de maximum</i>	107
Exercices	110
XI. Les tableaux à une dimension	111
1. <i>Le tableau : une façon commode de désigner plusieurs valeurs</i>	111
1.1. Quand la notion de variable ne suffit plus	111
1.2. La solution : le tableau	112
1.3. Un peu de vocabulaire	113
2. <i>Comment utiliser un tableau dans un programme</i>	114
2.1. Lui attribuer de la place et préciser son type	114
2.2. Employer une variable indicée comme une variable simple	115
2.3. Exemples d'affectation de valeurs à un tableau	115
2.4. Exemple de lecture des éléments d'un tableau	116
2.5. Exemple d'écriture des éléments d'un tableau	118
Exercices	118
2.6. Exemple d'utilisation de variables indicées dans des expres- sions	119
Exercices	120
3. <i>Quelques algorithmes classiques appliqués aux tableaux</i>	121
3.1. Calcul de la somme des éléments d'un tableau	121
Exercices	121
3.2. Détermination du plus grand élément d'un tableau	121
Exercices	123
4. <i>Lorsque l'on ne peut pas se passer d'un tableau</i>	125
XII. Les boucles imbriquées	127
1. <i>L'imbrication multiplie les tours de boucle</i>	128
1.1. Premier exemple : une boucle avec compteur dans une boucle jusqu'à	128
1.2. Deuxième exemple : deux boucles avec compteur imbri- quées	129
Exercices	130
2. <i>Exercice résolu : tables de multiplication</i>	130
Exercices	132

XIII. Les tableaux à deux dimensions	133
1. <i>Introduction</i>	133
2. <i>Comment utiliser un tableau à deux dimensions</i>	135
2.1. Lui attribuer de la place et préciser son type.....	135
2.2. Exemples d'affectation de valeurs	135
2.3. Exemples de lecture des éléments d'un tableau	136
2.4. Exemples d'écriture des éléments d'un tableau.....	137
Exercices	139
3. <i>Quelques algorithmes classiques appliqués aux tableaux à deux dimensions</i>	139
3.1. Calcul de la somme des éléments d'un tableau à deux dimensions	140
Exercices	141
3.2. Calcul du maximum des éléments d'un tableau à deux dimensions	141
Exercices	141
Corrections des exercices	143