

*Jean-Louis Roos*

*Intelligence  
artificielle en*  
**LANGAGE C**



# TABLE DES MATIERES

---

<b>CHAPITRE 1 - INTRODUCTION</b> .....	<b>1</b>
<b>CHAPITRE 2 - LES ENTIERS, LES CARACTERES ET LES POINTEURS</b> .....	<b>9</b>
1. LES ENTIERS ET LES CARACTERES .....	9
2. LES TABLEAUX D'ENTIERES ET DE CARACTERES .....	11
3. LES POINTEURS D'ENTIERES ET DE CARACTERES .....	12
3.1. Créer des pointeurs .....	13
3.2. Les pointeurs ne forment pas un type mais sont typés .....	14
3.3. Exercice: pointeur de pointeur! .....	14
3.4. Résumé .....	16
3.5. Tableaux et pointeurs .....	17
3.6. Affectation de pointeurs .....	18
4. COMPLEMENTS SUR LE LANGAGE C .....	19
4.1. Les fonctions .....	19
4.2. Le programme principal .....	22
4.3. Les opérateurs .....	22
4.4. Les structures de contrôle simples .....	24
4.5. Quelques fonctions C .....	25
5. UNE REPRESENTATION EN MEMOIRE INDEPENDANTE .....	26
5.1. Position du problème .....	26
5.2. Organisation générale de la mémoire dans CIA .....	27
5.3. Allocation de blocs de mémoire .....	27

<b>CHAPITRE 3 - LES STRUCTURES ET LES UNIONS</b> .....	<b>31</b>
1. LES STRUCTURES .....	31
1.1. Créer des structures .....	31
1.2. Manipulation d'éléments dans une structure .....	32
1.3. Structure de bits .....	33
1.4. Tableaux de structures et structures de structures .....	34
1.5. Alignement .....	35
1.6. Conversions .....	36
2. LES UNIONS .....	38
3. RETOUR SUR LA GESTION DE MEMOIRE DANS CIA .....	39
3.1. Objectifs .....	39
3.2. Déclarations .....	40
3.3. Autres choix possibles .....	41
3.4. Utilisation de portions de mémoire .....	41
3.5. Lecture de portions de mémoire .....	44
<b>CHAPITRE 4 - CHAINES DE CARACTERES ET SYMBOLES</b> .....	<b>47</b>
1. LES CHAINES DE CARACTERES .....	47
1.1. Les chaînes de caractères en langage C .....	48
1.2. Les principales fonctions de manipulation de chaînes .....	49
2. LE CONCEPT DE SYMBOLE DANS CIA .....	52
2.1. Structure d'un symbole .....	52
2.2. Dictionnaire de symboles .....	53
3. MANIPULER DES SYMBOLES .....	55
3.1. Principe général .....	55
3.2. Recherche dans le dictionnaire .....	57
3.3. Création dans le dictionnaire .....	58
<b>CHAPITRE 5 - C FACILE</b> .....	<b>61</b>
1. COMPLEMENTS SUR LE LANGAGE .....	61
1.1. Eléments lexicaux .....	61

1.2. Quelques définitions .....	62
1.3. Classes de mémorisation .....	63
1.4. Les ordres de traitement .....	68
1.5. La fonction de début de programme: main() .....	71
<b>2. LES LIBRAIRIES DU LANGAGE C .....</b>	<b>71</b>
2.1. Manipuler les caractères: ctype.h .....	72
2.2. Entrées et sorties: stdio.h .....	72
2.3. Librairie standard: stdlib.h .....	76
2.4. Librairie d'allocation malloc.h ou alloc.h .....	76
2.5. Manipuler des chaînes: string.h .....	77
2.6. Librairie mathématique: math.h .....	78
2.7. Divers: setjmp.h, time.h .....	78
<b>CHAPITRE 6 - LES TYPES ABSTRAITS DE CIA .....</b>	<b>79</b>
<b>1. LES TYPES DANS CIA .....</b>	<b>80</b>
1.1. Organisation des types .....	80
1.2. Détail des types .....	81
<b>2. LES TYPES RECONNUS EN LECTURE .....</b>	<b>87</b>
2.1. Les règles syntaxiques de CIA .....	87
2.2. La création des types abstraits .....	89
<b>3. LES TYPES RECONNUS EN TRAITEMENT .....</b>	<b>92</b>
3.1. Le traitement des nouveaux types .....	92
3.2. La création des nouveaux types .....	94
<b>CHAPITRE 7 - CREATION D'UN DICTIONNAIRE DE SYMBOLES .....</b>	<b>97</b>
<b>1. DEFINITION D'UNE FONCTION EXTERNE CIA .....</b>	<b>97</b>
1.1. Les fonctions initiales .....	98
1.2. Créer des fonctions dynamiquement .....	99
<b>2. CREATION D'UNE FONCTION CIA .....</b>	<b>100</b>
2.1. Déclarer la fonction écrite en C .....	101
2.2. Créer un symbole .....	101



2.3. Transformer un symbole en fonction .....	103
<b>3. LES AUTRES ELEMENTS DU DICTIONNAIRE .....</b>	<b>103</b>
3.1. Le symbole nil .....	104
3.2. Les symboles auto-évaluables .....	104
3.3. Les variables globales .....	104
3.4. Initialisation d'externes: les chemins vers nil, t, zéro .....	104
<b>4. L'INITIALISATION DE CIA .....</b>	<b>105</b>
<b>5. UTILISER uinit() .....</b>	<b>105</b>
5.1. Principe général .....	106
5.2. Exemple simple: la fonction lisse() .....	106
<b>CHAPITRE 8 - CONSTRUIRE UN INTERPRETEUR .....</b>	<b>109</b>
<b>1. GESTION DES ARGUMENTS .....</b>	<b>110</b>
1.1. Les arguments sont dans une FILE .....	110
1.2. Le contenu d'un argument .....	110
1.3. Déclarations .....	111
1.4. Schémas d'organisation .....	112
1.5. Gestion de la FILE .....	113
<b>2. LE LECTEUR .....</b>	<b>114</b>
2.1. La lecture du flux d'entrée dans une zone tampon .....	115
2.2. La codification .....	117
<b>3. L'EVALUATION .....</b>	<b>121</b>
3.1. Règles d'évaluation .....	121
3.2. Les deux évaluateurs .....	122
<b>4. L'IMPRESSION .....</b>	<b>123</b>
4.1. La fonction print1() .....	124
4.2. L'imprimeur .....	124
4.3. zprint1() .....	127
<b>5. LA GESTION DES ERREURS .....</b>	<b>127</b>
5.1. La table des messages d'erreur .....	127
5.2. L'enchaînement des programmes .....	128
<b>6. CIA ET LE TOPLEVEL .....</b>	<b>128</b>

6.1. La fonction main()	129
6.2. La boucle "read, eval, print"	131
6.3. Le toplevel	132
<b>7. FONCTIONNEMENT DE L'INTERPRETEUR</b>	<b>132</b>
7.1. Traitement des entités non fonctionnelles	132
7.2. Traitement des fonctions	132

**CHAPITRE 9 - LE PREPROCESSEUR** ..... **137**

<b>1. LES MACRO DEFINITIONS</b>	<b>137</b>
1.1. Les définitions de constantes	137
1.2. Les définitions de fonctions	138
<b>2. LES INCLUSIONS</b>	<b>138</b>
2.1. Le fichier d'inclusions ciah.h	139
2.2. Le fichier d'inclusions ciax.h	141
<b>3. LA COMPILATION CONDITIONNELLE</b>	<b>142</b>
3.1. Le "magic number"	142
3.2. Position des compilations conditionnelles	143
3.3. Exemples d'application avec Sas-C	144

**CHAPITRE 10 - LES UTILITAIRES POUR CREER DES FONCTIONS** ..... **147**

<b>1. LES UTILITAIRES TRES GENERAUX</b>	<b>147</b>
1.1. Les utilitaires liés au symboles	147
1.2. Utilitaire lié aux adresses	149
1.3. Utilitaires liés aux vecteurs	149
1.4. Création de chaînes de caractères	151
<b>2. LES SOUS-PROGRAMMES</b>	<b>152</b>
2.1. Analogie de deux vecteurs	152
2.2. Copie de séquences	153
2.3. Construire une sous-séquence	154
2.4. Longueur de séquences	156
2.5. Recherche dans une arborescence	157

2.6. Evaluation d'une chaîne ..... 158

**CHAPITRE 11 - LES FONCTIONS MATHÉMATIQUES ..... 159**

**1. REPRESENTATION DES NOMBRES ..... 159**

- 1.1. Création d'un chemin complet contenant un entier ..... 160
- 1.2. Création d'un "double" ..... 160
- 1.3. Traitement d'un nombre à partir du chemin complet ..... 161

**2. OPERATIONS MATHÉMATIQUES SIMPLES ..... 162**

**3. OPERATIONS VECTORIELLES ..... 164**

- 3.1. minimum et maximum d'un vecteur ..... 164
- 3.2. Cumul d'un vecteur de nombre ..... 166
- 3.3. Tri d'un vecteur: sort() ..... 167

**4. AUTRES OPERATIONS ..... 168**

- 4.1. Traitements de symboles: symbolvar() ..... 168
- 4.2. Nombre aléatoire: random() ..... 169
- 4.3. Evaluation d'une formule mathématique ..... 169

**5. LES PREDICATS ..... 173**

**CHAPITRE 12 - DETAIL DE QUELQUES FONCTIONS  
SIMPLES DANS CIA ..... 175**

**1. FONCTIONS SANS UTILITAIRES ..... 176**

- 1.1. Quelques fonctions élémentaires ..... 177
- 1.2. Quelques fonctions de "toplevel" ..... 180
- 1.3. Les prédicats simples ..... 183
- 1.4. Les fonctions sur symboles et sur chaînes ..... 185
- 1.5. Les fonctions simples avec longjmp() ..... 187

**2. LES FONCTIONS PLUS COMPLEXES ..... 188**

- 2.1. La liste des symboles ..... 188
- 2.2. Le typage ..... 189
- 2.3. Les prédicats complexes ..... 191

**3. LES FONCTIONS AVEC UTILITAIRES ..... 192**

- 3.1. Fonctions sur vecteurs ..... 192



3.2. Fonctions sur chaînes .....	194
----------------------------------	-----

**CHAPITRE 13 - ETUDE DE QUELQUES FONCTIONS COMPLEXES ..... 197**

<b>1. COMPILER .....</b>	<b>197</b>
1.1. La compilation .....	197
1.2. L'exécution .....	199
<b>2. LE PROCESSUS D'INFERENCE .....</b>	<b>201</b>
2.1. Règles et inférence, principe général .....	201
2.2. Créations de règles .....	201
2.3. Inférer .....	203
2.4. Le "break" .....	206
<b>3. GESTION DE TABLEAUX .....</b>	<b>206</b>
3.1. Définition .....	206
3.2. Impression .....	207
<b>4. LES FRAGMENTS .....</b>	<b>208</b>
4.1. Création de fragments .....	208
4.2. Utilisation de fragments .....	211
<b>5. FORMATER UNE CHAINE DE CARACTERES .....</b>	<b>212</b>
5.1. Le traitement principal du format .....	212
5.2. Les insertions et les conversions .....	213
<b>6. LES STRUCTURES DE CONTROLE .....</b>	<b>213</b>
6.1. La fonction ifx() .....	213
6.2. La fonction whilex .....	215
<b>7. LES APPLICATIONS AU SENS DE LISP .....</b>	<b>216</b>
7.1. La fonction map .....	216
7.2. map-array .....	218
7.3. map-setq .....	218

**CHAPITRE 14 - LES FICHIERS ..... 219**

<b>1. ORGANISATION GENERALE DE LA GESTION DE FICHIERS .....</b>	<b>219</b>
1.1. Table d'attachement .....	219