

Claude Delannoy

Programmer en langage C

Avec exercices corrigés

E Eyrolles

2-005-25-1

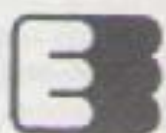
2-005-25-1

Programmer en langage C

Avec exercices corrigés

Claude DELANNOY

TROISIÈME TIRAGE
1999



Eyrolles

Table des matières

1 - GÉNÉRALITÉS SUR LE LANGAGE C.....	1
1 - PRÉSENTATION PAR L'EXEMPLE DE QUELQUES	
INSTRUCTIONS DU LANGAGE C.....	1
1.1 Un exemple de programme en langage C	1
1.2 Structure d'un programme en langage C	3
1.3 Déclarations	3
1.4 Pour écrire des informations : la fonction printf	4
1.5 Pour faire une répétition : l'instruction for	5
1.6 Pour lire des informations : la fonction scanf	6
1.7 Pour faire des choix : l'instruction if	7
1.8 Les directives à destination du préprocesseur	8
1.9 Un second exemple de programme	9
2 - QUELQUES RÈGLES D'ÉCRITURE	11
2.1 Les identificateurs	11
2.2 Les mots clés	11
2.3 Les séparateurs	12
2.4 Le format libre	13
2.5 Les commentaires	14
3 - CRÉATION D'UN PROGRAMME EN LANGAGE C	14
3.1 L'édition du programme	15
3.2 La compilation	15
3.3 L'édition de liens	15
3.4 Les fichiers en-tête	16

II. LES TYPES DE BASE DU LANGAGE C	17
1 - LA NOTION DE TYPE	17
2 - LES TYPES ENTIERS	19
2.1 Leur représentation en mémoire	19
2.2 Les différents types d'entiers	19
2.3 Notation des constantes entières	20
3 - LES TYPES FLOTTANTS	20
3.1 Les différents types et leur représentation en mémoire	20
3.2 Notation des constantes flottantes	21
4 - LES TYPES CARACTÈRES	22
4.1 La notion de caractère en langage C	22
4.2 Notation des constantes caractères	22
5 - INITIALISATION ET CONSTANTES	24
III. LES OPÉRATEURS ET LES EXPRESSIONS EN LANGAGE C	25
1 - L'ORIGINALITÉ DES NOTIONS D'OPÉRATEUR ET D'EXPRESSION EN LANGAGE C	25
2 - LES OPÉRATEURS ARITHMÉTIQUES EN C	27
2.1 Présentation des opérateurs	27
2.2 Les priorités relatives des opérateurs	28
3 - LES CONVERSIONS IMPLICITES POUVANT INTERVENIR DANS UN CALCUL D'EXPRESSION	29
3.1 Notion d'expression mixte	29
3.2 Les conversions d'ajustement de type	29
3.3 Les promotions numériques	30
3.4 Le cas du type char	31
4 - LES OPÉRATEURS RELATIONNELS	33
5 - LES OPÉRATEURS LOGIQUES	35
6 - L'OPÉRATEUR D'AFFECTATION ORDINAIRE	38
6.1 Notion de lvalue	38
6.2 L'opérateur d'affectation possède une associativité de droite à gauche	39
6.3 L'affectation peut entraîner une conversion	39
7 - LES OPÉRATEURS D'INCRÉMENTATION ET DE DÉCRÉMENTATION	40
7.1 Leur rôle	40
7.2 Leurs priorités	41
7.3 Leur intérêt	42
8 - LES OPÉRATEURS D'AFFECTATION ÉLARGIE	43
9 - LES CONVERSIONS FORCÉES PAR UNE AFFECTATION	44
10 - L'OPÉRATEUR DE CAST	45
11 - L'OPÉRATEUR CONDITIONNEL	46

12 - L'OPÉRATEUR SÉQUENTIEL	47
13 - L'OPÉRATEUR SIZEOF	49
14 - RÉCAPITULATIF DES PRIORITÉS DE TOUS LES OPÉRATEURS	49
EXERCICES	51
IV. LES ENTRÉES-SORTIES CONVERSATIONNELLES	53
1 - LES POSSIBILITÉS DE LA FONCTION PRINTF	54
1.1 Les principaux codes de conversion	54
1.2 Action sur le gabarit d'affichage	55
1.3 Actions sur la précision	56
1.4 La syntaxe de printf	57
1.5 En cas d'erreur de programmation	57
1.6 La macro putchar	58
2 - LES POSSIBILITÉS DE LA FONCTION SCANF	59
2.1 Les principaux codes de conversion de scanf	59
2.2 Premières notions de tampon et de séparateurs	60
2.3 Les premières règles utilisées par scanf	60
2.4 On peut imposer un gabarit maximal	62
2.5 Rôle d'un espace dans le format	62
2.6 Cas où un caractère invalide apparaît dans une donnée	62
2.7 Arrêt prématuré de scanf	63
2.8 La syntaxe de scanf	64
2.9 Problèmes de synchronisation entre l'écran et le clavier	65
2.10 En cas d'erreur	66
2.11 La macro getchar	68
EXERCICES	68
V. LES INSTRUCTIONS DE CONTRÔLE	71
1 - L'INSTRUCTION IF	72
1.1 Blocs d'instructions	72
1.2 Syntaxe de l'instruction if	73
1.3 Exemples	74
1.5 En cas d'imbrication des instructions if	75
2 - L'INSTRUCTION SWITCH	77
2.1 Exemples d'introduction de l'instruction switch	77
2.2 Syntaxe de l'instruction switch	80
3 - L'INSTRUCTION DO... WHILE	82
3.1 Exemple d'introduction de l'instruction do... while	82
3.2 Syntaxe de l'instruction do... while	83
3.3 Exemples	85

4 - L'INSTRUCTION WHILE.....	86
4.1 Exemple d'introduction de l'instruction while	86
4.2 Syntaxe de l'instruction while.....	87
5- L'INSTRUCTION FOR.....	88
5.1 Exemple d'introduction de l'instruction for.....	88
5.2 Syntaxe de l'instruction for.....	89
6 - LES INSTRUCTIONS DE BRANCHEMENT INCONDITIONNEL : BREAK, CONTINUE ET GOTO.....	92
6.1 L'instruction break	92
6.2 L'instruction continue	93
6.3 L'instruction goto.....	94
EXERCICES.....	95

VI. LA PROGRAMMATION MODULAIRE ET LES FONCTIONS.....97

1 - LA FONCTION : LA SEULE SORTE DE MODULE EXISTANT EN C.....	98
2 - EXEMPLE DE DÉFINITION ET D'UTILISATION D'UNE FONCTION EN C	99
3 - QUELQUES RÈGLES	102
3.1 Arguments muets et arguments effectifs.....	102
3.2 L'instruction return	102
3.3 Cas des fonctions sans valeur de retour ou sans arguments.....	103
3.4 Les anciennes formes de l'en-tête des fonctions	105
4 - LES FONCTIONS ET LEURS DÉCLARATIONS.....	105
4.1 Les différentes façons de déclarer (ou de ne pas déclarer) une fonction.....	105
4.2 Où placer la déclaration d'une fonction	107
4.3 A quoi sert la déclaration d'une fonction	107
5 - RETOUR SUR LES FICHIERS EN-TÊTE.....	108
6 - EN C, LES ARGUMENTS SONT TRANSMIS PAR VALEUR	109
7 - LES VARIABLES GLOBALES	111
7.1 Exemple d'utilisation de variables globales	111
7.2 La portée des variables globales	112
7.3 La classe d'allocation des variables globales	113
8 - LES VARIABLES LOCALES.....	113
8.1 La portée des variables locales.....	113
8.2 Les variables locales automatiques.....	114
8.3 Les variables locales statiques	114
8.4 Le cas des fonctions récursives.....	115
9 - LA COMPILATION SÉPARÉE ET SES CONSÉQUENCES	116
9.1 La portée d'une variable globale - la déclaration extern	117
9.2 Les variables globales et l'édition de liens.....	118
9.3 Les variables globales cachées - la déclaration static.....	119

10 - LES DIFFÉRENTS TYPES DE VARIABLES, LEUR PORTÉE ET LEUR CLASSE D'ALLOCATION	119
10.1 La portée des variables	120
10.2 Les classes d'allocation des variables	120
10.3 Tableau récapitulatif	121
11 - INITIALISATION DES VARIABLES	122
11.1 Les variables de classe statique	122
11.2 Les variables de classe automatique	122
EXERCICES	123
VII LES TABLEAUX ET LES POINTEURS	125
1 - LES TABLEAUX À UN INDICE	125
1.1 Exemple d'utilisation d'un tableau en C	125
1.2 Quelques règles	127
2 - LES TABLEAUX À PLUSIEURS INDICES	128
2.1 Leur déclaration	128
2.2 Arrangement en mémoire des tableaux à plusieurs indices	129
3 - INITIALISATION DES TABLEAUX	130
3.1 Initialisation de tableaux à un indice	130
3.2 Initialisation de tableaux à plusieurs indices	131
4 - NOTION DE POINTEUR - LES OPÉRATEURS * ET &	131
4.1 Introduction	131
4.2 Quelques exemples	133
4.3 Incrémentation de pointeurs	134
5 - COMMENT SIMULER UNE TRANSMISSION PAR ADRESSE AVEC UN POINTEUR	135
6 - UN NOM DE TABLEAU EST UN POINTEUR CONSTANT	137
6.1 Cas des tableaux à un indice	137
6.2 Cas des tableaux à plusieurs indices	139
7 - LES OPÉRATIONS RÉALISABLES SUR DES POINTEURS	140
7.1 La comparaison de pointeurs	140
7.2 La soustraction de pointeurs	141
7.3 Les affectations de pointeurs et le pointeur nul	141
7.4 Les conversions de pointeurs	141
7.5 Les pointeurs génériques	142
8 - LES TABLEAUX TRANSMIS EN ARGUMENT	144
8.1 Cas des tableaux à un indice	144
8.2 Cas des tableaux à plusieurs indices	146
9 - UTILISATION DE POINTEURS SUR DES FONCTIONS	149
9.1 Paramétrage d'appel de fonctions	149
9.2 Fonctions transmises en argument	150
EXERCICES	151

VIII. LES CHAÎNES DE CARACTÈRES	15
1 - REPRÉSENTATION DES CHAÎNES	15
1.1 La convention adoptée	15
1.2 Cas des chaînes constantes	15
1.3 Initialisation de tableaux de caractères	15
1.4 Initialisation de tableaux de pointeurs sur des chaînes	15
2 - POUR LIRE ET ÉCRIRE DES CHAÎNES	15
3 - POUR FIABILISER LA LECTURE AU CLAVIER : LE COUPLE GETS - SCANF	16
4 - GÉNÉRALITÉS SUR LES FONCTIONS PORTANT SUR DES CHAÎNES	16
4.1 Ces fonctions travaillent toujours sur des adresses	16
4.2 La fonction <i>strlen</i>	16
4.3 Le cas des fonctions de concaténation	16
5 - LES FONCTIONS DE CONCATÉNATION DE CHAÎNES	16
5.1 La fonction <i>strcat</i>	16
5.2 La fonction <i>strncat</i>	16
6 - LES FONCTIONS DE COMPARAISON DE CHAÎNES	16
7 - LES FONCTIONS DE COPIE DE CHAÎNES	16
8 - LES FONCTIONS DE RECHERCHE DANS UNE CHAÎNE	16
9 - LES FONCTIONS DE CONVERSION	16
10 - QUELQUES PRÉCAUTIONS À PRENDRE AVEC LES CHAÎNES	16
10.1 Une chaîne possède une vraie fin, mais pas de vrai début	16
10.2 Les risques de modification des chaînes constantes	17
EXERCICES	17
IX. LES STRUCTURES	173
1 - DÉCLARATION D'UNE STRUCTURE	173
2 - UTILISATION D'UNE STRUCTURE	174
2.1 Utilisation des champs d'une structure	174
2.2 Utilisation globale d'une structure	175
2.3 Initialisations de structures	176
3 - POUR SIMPLIFIER LA DÉCLARATION DE TYPES : DÉFINIR DES SYNONYMES AVEC <i>TYPDEF</i>	176
3.1 Exemples d'utilisation de <i>typedef</i>	177
3.2 Application aux structures	177
4 - IMBRICATION DE STRUCTURES	178
4.1 Structure comportant des tableaux	178
4.2 Tableaux de structures	180
4.3 Structures comportant d'autres structures	181
5 - À PROPOS DE LA PORTÉE DU MODÈLE DE STRUCTURE	182

6 - TRANSMISSION D'UNE STRUCTURE EN ARGUMENT D'UNE FONCTION	183
6.1 Transmission de la valeur d'une structure	183
6.2 Transmission de l'adresse d'une structure : l'opérateur ->	184
7 - TRANSMISSION D'UNE STRUCTURE EN VALEUR DE RETOUR D'UNE FONCTION	185
EXERCICES	186
X. LES FICHIERS	187
1 - CRÉATION SÉQUENTIELLE D'UN FICHIER	188
2 - LISTE SÉQUENTIELLE D'UN FICHIER	190
3 - L'ACCÈS DIRECT	192
3.1 Accès direct en lecture sur un fichier existant	192
3.2 Les possibilités de l'accès direct	193
3.3 En cas d'erreur	195
4 - LES ENTRÉES-SORTIES FORMATÉES ET LES FICHIERS DE TEXTE	196
5 - LES DIFFÉRENTES POSSIBILITÉS D'OUVERTURE D'UN FICHIER	198
6 - LES FICHIERS PRÉDÉFINIS	199
EXERCICES	201
XI. LA GESTION DYNAMIQUE	203
1 - LES OUTILS DE BASE DE LA GESTION DYNAMIQUE : MALLOC ET FREE	204
1.1 La fonction malloc	204
1.2 La fonction free	206
2 - D'AUTRES OUTILS DE GESTION DYNAMIQUE : CALLOC ET REALLOC	208
2.1 La fonction calloc	208
2.2 La fonction realloc	209
3 - EXEMPLE D'APPLICATION DE LA GESTION DYNAMIQUE : CRÉATION D'UNE LISTE CHAÎNÉE	209
EXERCICE	212
XII. LE PRÉPROCESSEUR	213
1 - LA DIRECTIVE #INCLUDE	213
2 - LA DIRECTIVE #DEFINE	214
2.1 Définition de symboles	215
2.2 Définition de macros	217

3 - LA COMPILATION CONDITIONNELLE	220
3.1 Incorporation liée à l'existence de symboles	220
3.2 Incorporation liée à la valeur d'une expression	221
XIII. LES POSSIBILITÉS DU LANGAGE C PROCHES DE LA MACHINE	223
1 - COMPLÉMENTS SUR LES TYPES D'ENTRIERS	223
1.1 Rappels concernant la représentation des nombres entiers en binaire	223
1.2 Prise en compte d'un attribut de signe	225
1.3 Extension des règles de conversions	225
1.4 La notation octale ou hexadécimale des constantes	226
2 - COMPLÉMENTS SUR LES TYPES DE CARACTÈRES	226
2.1 Prise en compte d'un attribut de signe	226
2.2 Extension des règles de conversion	228
3 - LES OPÉRATEURS DE MANIPULATION DE BITS	228
3.1 Présentation des opérateurs de manipulation de bits	228
3.2 Les opérateurs bit à bit	229
3.3 Les opérateurs de décalage	230
3.4 Exemples d'utilisation des opérateurs de bits	231
4 - LES CHAMPS DE BITS	232
5 - LES UNIONS	233
ANNEXE : LES PRINCIPALES FONCTIONS DE LA BIBLIOTHEQUE STANDARD	237
1 - ENTRÉES-SORTIES (STDIO.H)	238
1.2 Ecriture formatée	238
1.3 Lecture formatée	243
1.4 Entrées-sorties de caractères	246
1.5 Entrées-sorties sans formatage	248
1.6 Action sur le pointeur de fichier	249
1.7 Gestion des erreurs	249
2 - TESTS DE CARACTÈRES ET CONVERSIONS MAJUSCULES-MINUSCULES (CTYPE.H)	250
4 - FONCTIONS MATHÉMATIQUES (MATH.H)	253
5 - UTILITAIRES (STDLIB.H)	254
CORRECTION DES EXERCICES	257
INDEX	275