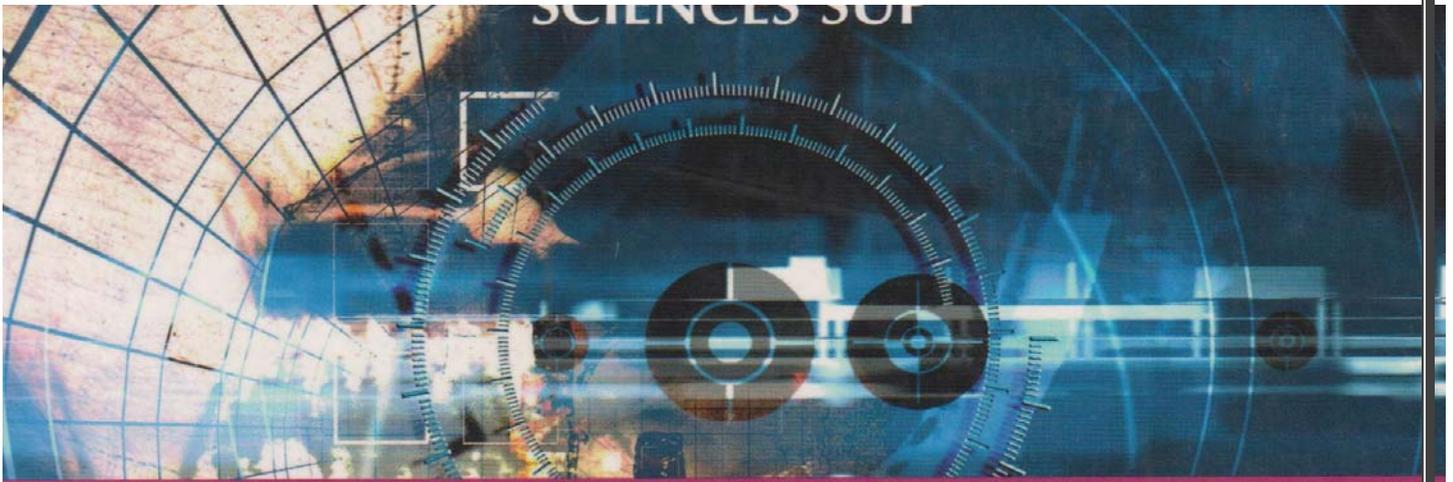


SCIENCES SUP



Cours et exercices corrigés

IUT • BTS • Licence • Écoles d'ingénieurs

INITIATION À L'ALGORITHMIQUE ET AUX STRUCTURES DE DONNÉES EN C

118 exercices corrigés



*Rémy Malgouyres
Rita Zrour
Fabien Feschet*

DUNOD

2-005-653-2

INITIATION À L'ALGORITHMIQUE ET AUX STRUCTURES DE DONNÉES



EN C

118 exercices corrigés

Rémy Malgouyres

Professeur à l'université de Clermont 1

Rita Zrour

Docteur en informatique et ATER à l'université de Clermont 1

Fabien Feschet

Professeur à l'université de Clermont 1

DUNOD

Table des matières

AVANT-PROPOS

iii

PARTIE I • BASES DU LANGAGE C

CHAPITRE 1 • QU'EST-CE QU'UN ORDINATEUR ?

2

1.1 Exemples d'applications de l'informatique

2

1.2 Codage des données

2

1.3 Fonctionnement d'un ordinateur

3

CHAPITRE 2 • PREMIERS PROGRAMMES

6

2.1 Qu'est-ce qu'un programme ?

6

2.2 Afficher un mot

7

2.3 Lire un nombre

7

2.4 Effectuer un calcul et mémoriser le résultat

8

Exercices

9

CHAPITRE 3 • TYPES DE DONNÉES

10

3.1 Variables et opérations

10

3.2 Type entier int

10

3.3 Les types réels float et double

11

3.4 Le type char

12

3.5 Les types unsigned

12

3.6 Affectations et conversions

12

3.7 Les constantes et le #define

13

3.8 Définir ses propres types

14

Exercices

15

CHAPITRE 4 • ENTRÉES-SORTIES : stdio.h

16

4.1 Qu'est-ce qu'une bibliothèque d'entrées-sorties ?

16

4.2	L'affichage de données sous forme de texte	17
4.3	Lecture au clavier	18
	Exercices	19
	CHAPITRE 5 • EXÉCUTION CONDITIONNELLE	20
5.1	Qu'est-ce que l'exécution conditionnelle ?	20
5.2	Condition si – alors	20
5.3	Condition si – alors – sinon	21
5.4	Notions de calcul booléen	22
5.5	Le switch	24
	Exercices	26
	CHAPITRE 6 • STRUCTURATION D'UN PROGRAMME C	27
6.1	Qu'est-ce qu'un sous-programme ?	27
6.2	Exemple de fonction C	27
6.3	Exemple de structuration d'un programme	28
6.4	Forme générale d'une fonction C	30
6.5	Passage de paramètres par valeur	30
	Exercices	31
	CHAPITRE 7 • STRUCTURES	31
7.1	Déclaration d'une structure	31
7.2	Utilisation d'une structure	31
	Exercices	35
	CHAPITRE 8 • ITÉRATION	37
8.1	Boucle while	37
8.2	Boucle for	38
	Exercices	39
	PARTIE II • STRUCTURES SÉQUENTIELLES	
	CHAPITRE 9 • TABLEAUX	42
9.1	Déclaration d'un tableau	42
9.2	Accès aux éléments	42
9.3	Nombre d'éléments fixé	42
9.4	Nombre d'éléments variable borné	42
9.5	Initialisation lors de la déclaration	42
	Exercices	44

CHAPITRE 10 • FICHIERS TEXTE	47
10.1 Qu'est-ce qu'un fichier texte ?	47
10.2* Ouverture et fermeture d'un fichier texte	47
10.3 Lire et écrire des données formatées	49
Exercices	53
CHAPITRE 11 • ADRESSES, POINTEURS ET PASSAGE PAR ADRESSE	54
11.1 Mémoire centrale et adresses	54
11.2 Variables de type pointeur	54
11.3 Passage de paramètre par valeur	56
11.4 Passage de paramètre par adresse	56
Exercices	58
CHAPITRE 12 • ALLOCATION DYNAMIQUE	59
12.1 Gestion de la mémoire centrale	59
12.2 Allocation avec malloc	60
12.3 Allocation avec calloc	61
Exercices	63
CHAPITRE 13 • CHAÎNES DE CARACTÈRES	65
13.1 Qu'est-ce qu'une chaîne de caractères ?	65
13.2 Opérations prédéfinies sur les chaînes	66
Exercices	71
CHAPITRE 14 • FICHIERS BINAIRES	72
14.1 Différence entre fichiers texte et binaire	72
14.2 Ouverture et fermeture d'un fichier binaire	72
14.3 Lecture dans un fichier binaire	73
14.4 Écriture dans un fichier binaire	75
14.5 Se positionner dans un fichier binaire	76
Exercices	76
CHAPITRE 15 • TABLEAUX À DOUBLE ENTRÉE	78
15.1 Tableaux de dimension 2	78
15.2 Allocation dynamique et libération d'un tableau de dimension 2	79
Exercices	81

PARTIE III • ALGORITHMES**CHAPITRE 16 • LANGAGE ALGORITHMIQUE ET COMPLEXITÉ**

16.1 Pourquoi un autre langage ? 8

16.2 Types 8

16.3 Entrées-sorties 8

16.4 Syntaxe 8

16.5 Fonctions et procédures 8

16.6 Enregistrements 8

16.7 Pointeurs, adresses et allocation 8

16.8 Notion de complexité d'un algorithme 8

Exercices 9

CHAPITRE 17 • ALGORITHMES DE TRI QUADRATIQUES

17.1 Qu'est-ce qu'un tri ? 93

17.2 Tri par sélection 93

17.3 Tri par insertion 95

17.4 Tri par bulles 96

CHAPITRE 18 • LE TRI RAPIDE (QUICKSORT)

18.1 Partitionnement 98

18.2 L'algorithme de tri rapide 99

18.3 Comparaison de temps de calcul 100

Exercices 101

PARTIE IV • STRUCTURES DE DONNÉES**CHAPITRE 19 • LISTES CHÂÎNÉES**

19.1 Qu'est-ce qu'une liste chaînée ? 104

19.2 Déclarer une liste chaînée 105

19.3 Insertion en tête de liste 106

19.4 Construction d'une liste chaînée 106

19.5 Parcours de liste 107

19.6 Insertion en queue de liste 108

19.7 Libération de mémoire 110

Exercices 111

CHAPITRE 20 • PILES	113
20.1 Qu'est-ce qu'une pile ?	113
20.2 Implémentation sous forme de tableau	114
20.3 Implémentation sous forme de liste chaînée	116
20.4 Comparaison entre tableaux et listes chaînées	118
Exercices	119
CHAPITRE 21 • FILES	120
21.1 Qu'est-ce qu'une file ?	120
21.2 Gestion naïve par tableaux	121
21.3 Gestion circulaire par tableaux	122
21.4 Gestion par listes chaînées	124
Exercice	126
CHAPITRE 22 • RÉCURSIVITÉ	127
22.1 Qu'est-ce que la récursivité ?	127
22.2 Comment programmer une fonction récursive ?	128
22.3 Pile d'appels	128
Exercices	129
CHAPITRE 23 • ARBRES BINAIRES	131
23.1 Qu'est-ce qu'un arbre binaire ?	131
23.2 Parcours d'arbres binaires	132
23.3 Libération de mémoire	135
Exercices	136
CHAPITRE 24 • GRAPHERS	137
24.1 Définition mathématique d'un graphe	137
24.2 Chemins dans un graphe	138
24.3 Représentation par matrices d'adjacence	138
Exercices	140
CHAPITRE 25 • PARCOURS DE GRAPHERS	141
25.1 Parcours en profondeur récursif	141
25.2 Parcours en largeur	142

CHAPITRE 26 • LISTES D'ADJACENCE	145
26.1 Représentation par listes d'adjacence	145
Exercices	146
CORRIGÉS DES EXERCICES	148
ANNEXES	227
ANNEXE A • NOTIONS SUR LA COMPILATION	228
A.1 Qu'est-ce qu'un compilateur C ANSI ?	228
A.2 Compiler son premier programme	228
ANNEXE B • PROGRAMMATION MULTIFICHIERS	230
B.1 Mettre du code dans plusieurs fichiers	230
B.2 Compiler un projet multifichiers	231
ANNEXE C • COMPLÉMENTS SUR LE LANGAGE C	233
C.1 Énumérations	233
C.2 Unions	234
C.3 Variables globales	235
C.4 Do...while	235
C.5 <code>i++</code> et <code>++i</code>	236
C.6 Le générateur aléatoire : fonction <code>rand</code>	236
C.7 <code>break</code> et <code>continue</code>	237
C.8 Macros	238
C.9 <code>atoi</code> , <code>sprintf</code> et <code>sscanf</code>	238
C.10 <code>fgetc</code> et <code>fputc</code>	240
C.11 Arithmétique de pointeurs	241
INDEX	243