

Table des matières

I	Programmation	1
1	Récurtivité	3
1.1	Problème : la somme des n premiers entiers	3
1.2	Formulations récursives	6
1.3	Programmer avec des fonctions récursives	11
	Exercices	16
2	Modularité	19
2.1	Variations sur les ensembles	19
2.2	Modules, interfaces et encapsulation	26
2.3	Exceptions	34
	Exercices	40
3	Programmation objet	45
3.1	Classes et attributs : structurer les données	46
3.2	Méthodes : manipuler les données	50
3.3	Retour sur l'encapsulation	57
3.4	Héritage	59
	Exercices	62
4	Mise au point des programmes	71
4.1	Types	71
4.2	Tester un programme	79
4.3	Invariant de structure	84
	Exercices	86
5	Programmation fonctionnelle	89
5.1	Fonctions passées en arguments	91
5.2	Fonctions renvoyées comme résultats	95
5.3	Structures de données immuables	97
	Exercices	100

II	Algorithmique	105
6	Listes chaînées	107
6.1	Structure de liste chaînée	108
6.2	Opérations sur les listes	111
6.3	Modification d'une liste	119
6.4	Encapsulation dans un objet	121
	Exercices	124
7	Piles et files	127
7.1	Interface et utilisation d'une pile	128
7.2	Interface et utilisation d'une file	130
7.3	Réaliser une pile avec une liste chaînée	133
7.4	Réaliser une file avec une liste chaînée mutable	135
7.5	Réaliser une file avec deux piles	138
	Exercices	141
8	Arbres binaires	147
8.1	Structures arborescentes	147
8.2	Définition et propriétés des arbres binaires	148
8.3	Représentation en Python	151
8.4	Algorithmique des arbres binaires	153
	Exercices	155
9	Arbres binaires de recherche	157
9.1	Notion	158
9.2	Recherche dans un ABR	159
9.3	Ajout dans un ABR	161
9.4	Suppression dans un ABR	166
9.5	Encapsulation dans un objet	168
9.6	Arbre équilibré	169
	Exercices	170
10	Autres structures arborescentes	173
10.1	Arborescence	173
10.2	Exemples de structures arborescentes	176
	Exercices	185
11	Graphes	189
11.1	Définitions	190
11.2	Exemples de graphes	194
11.3	Représentation d'un graphe en Python	196
11.4	Exemple d'algorithme sur les graphes	202
	Exercices	204

12 Parcours en profondeur et en largeur	207
12.1 Parcours en profondeur	208
12.2 Détecter la présence d'un cycle dans un graphe orienté	212
12.3 Parcours en largeur	214
Exercices	218
13 Diviser pour régner	221
13.1 Retour sur la recherche dichotomique	222
13.2 Tri fusion	224
Exercices	229
14 Programmation dynamique	233
14.1 Rendu de monnaie	234
14.2 Alignement de séquences	239
Exercices	246
15 Recherche textuelle	249
15.1 Un algorithme simple	250
15.2 Accélération de la recherche	252
15.3 Un algorithme plus efficace : Boyer-Moore	254
Exercices	261
16 Calculabilité	263
16.1 Problème : détecteur d'erreurs en Python	263
16.2 Calculabilité, des mathématiques à l'informatique	268
16.3 Limites de la calculabilité	274
Exercices	280
III Bases de données	283
17 Modèle relationnel	285
17.1 Le modèle relationnel	286
17.2 Modélisation relationnelle des données	288
Exercices	295
18 Bases de données relationnelles	297
18.1 SQL : un langage de définition de données	298
18.2 Types de données en SQL	300
18.3 Spécification des contraintes d'intégrité	302
18.4 Suppression de tables	304
18.5 Insertion dans une table	305
Exercices	306

19 Requêtes SQL et mises à jour	309
19.1 Sélection de données	310
19.2 Modification des données	319
19.3 Requêtes imbriquées	323
Exercices	325
20 Systèmes de Gestion de Bases de Données	329
20.1 Historique	330
20.2 Transactions	331
20.3 Interaction entre un SGBD et un programme	337
Exercices	342
IV Architectures matérielles, systèmes d'exploitation et réseaux	345
21 Circuits intégrés	347
21.1 Microcontrôleurs	348
21.2 Système sur puce	351
Exercices	356
22 Gestion des processus et des ressources	359
22.1 L'ordonnanceur	360
22.2 Commandes Unix de gestion des processus	364
22.3 Programmation concurrente	367
Exercices	378
23 Protocoles de routage	383
23.1 Le protocole RIP	386
23.2 Protocole OSPF	390
23.3 Commandes système	396
Exercices	402
24 Sécurisation des communications	405
24.1 Cryptographie symétrique	406
24.2 Cryptographie asymétrique	410
24.3 Authentification des participants	415
24.4 Le protocole HTTPS	417
Exercices	424
Solutions des exercices	427
Index	507