

**Alain Cardon**

**Christophe Dabancourt**

**Classes préparatoires  
BTS - Deug - IUP - IUT**

# **Initiation à l'algorithmique objet**

**Modélisation avec UML et  
exemples de code en Java et C++**

**Avec 35  
exercices  
corrigés**

**E** Eyrolles

005-311-1

2-005-311-1

# Initiation à l'algorithmique objet

**Modélisation avec UML et  
exemples de code en Java et C++**

Alain Cardon  
Christophe Dabancourt

Deuxième tirage 2001

 **Eyrolles**

# Table des matières

---

## PARTIE I

---

<b>Notions de base</b> .....	1
CHAPITRE 1	
<b>Algorithmes et programmes</b> .....	3
<b>Informatique et algorithmique</b> .....	3
Le calculable .....	4
Ordinateur et machine programmable abstraite.....	4
<b>Les niveaux dans l'appréhension des traitements</b> .....	8
<b>Instructions algorithmiques, instructions machine et mémoire</b> ..	9
Étapes de réalisation d'un programme .....	9
Niveau transfert-registre et adressage .....	10
<b>Les définitions fondamentales</b> .....	13
CHAPITRE 2	
<b>Le langage algorithmique</b> .....	17
<b>Types primitifs et déclarations de variables</b> .....	17
Types de données.....	18
Caractères, indications de titre et de mise en forme .....	18
Les types primitifs numériques .....	19
Le type logique .....	20
Le type caractère .....	20
La déclaration des variables.....	21
<b>Instruction d'affectation et expression</b> .....	22
Délimiteur de bloc d'instructions .....	23

<b>Instructions de séquençement et de répétition</b> .....	24
Séquençement .....	24
Répétition .....	26
<b>Instructions d'entrée-sortie</b> .....	29
<b>Le type primitif tableau</b> .....	31
<b>Résumé des instructions utilisées</b> .....	33
<b>Programmes d'application</b> .....	33
Affectation .....	34
Condition .....	34
Répétition .....	38
Entrée-sortie .....	39
Tableau .....	40
<b>CHAPITRE 3</b> .....	43
<b>Les fonctions</b> .....	43
<b>Une méthode de décomposition d'un problème</b> .....	43
<b>Définition syntaxique d'une fonction et de son appel</b> .....	44
<b>Exécution d'une fonction et environnement</b> .....	46
Portée des variables .....	48
Passage des paramètres .....	50
Limitations d'usage des fonctions .....	55
<b>Programmes d'application</b> .....	55
Fonction sans retour de valeur .....	55
Fonction retournant une valeur .....	57
Portée des variables .....	57
Passage par valeur .....	59
Passage des références des objets par valeur .....	60
<b>CHAPITRE 4</b> .....	65
<b>Projet : les algorithmes de base</b> .....	65
<b>Utilisation de l'affectation</b> .....	65
Facture téléphonique .....	65
Quelques pièces .....	66
<b>Utilisation de la conditionnelle</b> .....	67
Location d'une voiture .....	67
Saisir un mois .....	68
<b>Utilisation des boucles</b> .....	69
Afficher des nombres .....	69
Saisir un mois .....	70
Saisir un mois avec cinq essais au maximum .....	70

<b>Utilisation de fonctions</b> .....	71
Facture téléphonique .....	71
Passage par référence .....	72
<b>Programmes d'application</b> .....	75
Facture téléphonique .....	75
Quelques pièces .....	76
Location .....	77
Afficher une suite de nombres .....	78
Calcul du maximum de deux entiers .....	79
Saisie d'un mois .....	81
<b>PARTIE II</b>	
<hr/>	
<b>Objets et classes</b> .....	85
CHAPITRE 5	
<b>Les objets</b> .....	87
<b>Les phases de développement d'un programme</b> .....	87
<b>L'approche objet</b> .....	88
<b>Objets, attributs et méthodes</b> .....	90
<b>Notions de classe et d'instance</b> .....	94
Exemple .....	96
Déclaration d'une classe .....	97
Déclaration et création d'une instance d'une classe .....	97
Définition externe d'une méthode : opérateur de portée .....	98
Passage des arguments des méthodes .....	100
Invocation d'une méthode .....	101
Utilisation de la classe Personne .....	102
Les constructeurs .....	103
<b>Programmes d'application</b> .....	104
La classe SommeN .....	104
La classe Personne .....	107
CHAPITRE 6	
<b>Analyse et conception objet</b> .....	111
<b>Méthodologie en approche objet</b> .....	111
Analyse .....	112
Conception .....	112
Classes d'objets et encapsulation .....	116

<b>Notation UML d'une classe d'objets</b> .....	117
Protection .....	118
Schéma réduit d'une classe .....	119
<b>L'héritage</b> .....	119
Héritage et graphe d'héritage .....	119
Exemple .....	120
Définition des classes : notations .....	122
<b>Liens et agrégation</b> .....	123
<b>Le polymorphisme</b> .....	125
<b>Surcharge et redéfinition</b> .....	126
<b>Extensions : héritage multiple et pseudo-variables this et super</b> .	126
Héritage multiple .....	126
Masquage .....	126
Pseudo-variables this et super .....	128
<b>Programmes d'application</b> .....	129
Héritage .....	129
<b>CHAPITRE 7</b>	
<b>Projet : les premières classes</b> .....	135
<b>La classe Chaine</b> .....	135
Présentation de la classe Chaine .....	135
Inversion d'une chaîne .....	137
Remplacement de caractères dans une chaîne .....	139
Exemple des départements .....	140
<b>Algorithmes de modification de la classe Chaine</b> .....	142
Recherche de motifs .....	142
Transformation d'une chaîne de caractères en entiers .....	143
<b>La classe Date</b> .....	146
Présentation de la classe Date .....	146
Écriture de la classe Date .....	149
Autres méthodes de la classe Date .....	153
<b>Exercice complémentaire</b> .....	154
<b>Programmes d'application</b> .....	155
Inversion d'une chaîne .....	155
Remplacement d'un caractère dans une chaîne .....	158
Application de la classe Chaine : les départements .....	160
Modification de la classe Chaine : nouvelles méthodes .....	160
Création de la classe Date .....	164

## PARTIE III

<b>Les algorithmes fondamentaux</b> .....	169
CHAPITRE 8	
<b>Collections, tableaux et algorithmes de tri</b> .....	171
<b>Collections et classe Collection</b> .....	171
<b>Les tableaux</b> .....	173
Définition et caractères d'un tableau.....	173
La classe Tableau.....	175
Utilisation de la classe Tableau.....	176
<b>Algorithmes de tri</b> .....	181
La classe TableauReelTrie.....	182
Le tri par le max.....	185
Le tri à bulles.....	186
<b>Recherche dichotomique</b> .....	189
<b>Complexité des algorithmes</b> .....	191
Opération élémentaire et complexité.....	191
Complexité en temps dans le pire des cas.....	192
<b>Exemple de la matrice 3x3</b> .....	196
<b>Programmes d'application</b> .....	197
La classe Tableau.....	197
Les classes Etudiant et Université.....	197
La classe TableauReelTrie.....	202
Exemple de complexité.....	208
CHAPITRE 9	
<b>La récursivité</b> .....	211
<b>Notion de récursivité</b> .....	211
<b>La fonction factorielle</b> .....	212
<b>La suite de Fibonacci</b> .....	214
<b>Inversion d'une chaîne de caractères par une méthode récursive</b> .....	215
<b>Exercice : le tri rapide</b> .....	217
<b>Programmes d'application</b> .....	221
Calcul de Factorielle .....	221
Calcul des éléments d'une suite.....	221
Inversion d'une chaîne de caractères.....	223

## CHAPITRE 10

<b>Projet : tableaux et récursivité</b> .....	225
<b>La classe TableauReel</b> .....	225
Construire la classe TableauReel .....	225
Les méthodes de la classe TableauReel .....	227
<b>La classe Dictionnaire</b> .....	233
La classe Définition .....	234
La classe Dictionnaire .....	235
<b>Récursivité : dessin d'une hiérarchie équilibrée</b> .....	238
Utilisation de la hiérarchie des classes de Java 1.2 .....	240
Classe Hierarchie .....	241
<b>Programmes d'application</b> .....	242
Création de la classe TableauReel .....	242
Création de la classe Dictionnaire .....	250
Création de la classe Hierarchie .....	255

## PARTIE IV

---

**Les structures dynamiques** ..... 257

## CHAPITRE 11

<b>Collections et listes</b> .....	259
<b>Les listes</b> .....	259
Notion de liste .....	259
Manipulation des listes comme objets .....	260
La classe Liste .....	262
La classe ListeChainée .....	263
Utilisation de la classe ListeChainée .....	264
<b>Les piles</b> .....	265
Collection et pile .....	266
La classe Pile .....	267
Exemple d'utilisation de la classe Pile .....	268
<b>Les arbres binaires</b> .....	270
Collection et arbres binaires .....	270
Représentation d'un arbre binaire .....	272
La classe Arbre .....	272
Exemple d'utilisation de la classe ArbreBinaire .....	273
<b>Programmes d'application</b> .....	274
Utilisation de la classe Liste .....	274



Utilisation de la classe Pile : la calculatrice. . . . .	276
<b>CHAPITRE 12</b>	
<b>Projet : les structures dynamiques</b> . . . . .	279
<b>Implémentation des structures chaînées</b> . . . . .	279
Schémas des cellules et des structures chaînées . . . . .	280
Schéma d'une cellule. . . . .	280
La classe CelluleEntier . . . . .	281
La classe ListeChaine . . . . .	282
La pile . . . . .	285
<b>Les arbres binaires de recherche</b> . . . . .	287
Notion formelle . . . . .	287
Parcours d'un arbre . . . . .	288
La classe ArbreBinaireRecherche . . . . .	291
<b>Autres structures chaînées</b> . . . . .	296
Liste circulaire . . . . .	296
File FIFO (First In, First Out) . . . . .	296
Liste ordonnée . . . . .	297
<b>Exercices complémentaires</b> . . . . .	299
<b>Programmes d'application</b> . . . . .	299
La classe CelluleEntier . . . . .	299
La classe ListeChaine . . . . .	300
La classe PileEntier . . . . .	303
La classe Noeud. . . . .	304
Les classes Liste, Cellule et ListeOrdonnée . . . . .	307

## PARTIE V

<b>Processus et événements</b> . . . . .	313
<b>CHAPITRE 13</b>	
<b>Les processus</b> . . . . .	315
<b>Le système d'exploitation</b> . . . . .	315
Le noyau du système . . . . .	316
Les processus. . . . .	317
Correspondance programmes-processus . . . . .	317
États d'un processus . . . . .	318
Priorité d'un processus . . . . .	319
Communication et synchronisation entre processus :	
les sémaphores. . . . .	320

<b>Approche objet des processus</b> .....	322
La classe Scheduler .....	322
La classe Process .....	322
La classe ListeProcess .....	324
<b>Approche objet des sémaphores</b> .....	325
<b>Programmes d'applications</b> .....	326
<b>CHAPITRE 14</b>	
<b>Les événements</b> .....	329
<b>Des objets autonomes grâce aux événements</b> .....	329
Les événements sont des objets .....	330
Liste des écouteurs .....	330
<b>Exemple de gestion d'événement en Java</b> .....	331
<b>CHAPITRE 15</b>	
<b>Projet : le Démineur</b> .....	335
<b>Règles du Démineur</b> .....	335
<b>Première analyse : les classes de base</b> .....	336
La classe Demineur1 .....	337
La classe Terrain .....	337
La classe Case .....	338
La classe GestionSouris .....	339
La classe Hasard .....	340
Bilan .....	341
<b>Approfondissement de l'analyse : quelques événements</b> .....	341
La classe Case mine .....	342
La classe Explosion .....	342
La classe Demineur2 .....	343
Terminer la partie .....	343
<b>Une analyse encore plus fine</b> .....	343
La classe Demineur3 .....	344
Bouton et Label .....	344
La classe Horloge .....	344
La fin de partie .....	345
Bilan du projet .....	346
<b>Programmes d'application</b> .....	346
Commençons par définir les cases qui composent le terrain .....	346
Les mines rendent le jeu plus aléatoire en spécialisant les Cases ..	347
Le jeu doit être lancé dans une fenêtre .....	348
La classe Terrain est l'objet pivot du jeu .....	349

Si une mine explose, elles doivent toutes exploser . . . . .	351
Gestion de la souris : implémentons l'écouteur MouseListener . .	352
Le hasard permet de placer les mines sur le terrain. . . . .	353
Pour gérer un compteur de temps indépendant du jeu : créons un nouveau processus. . . . .	354

## ANNEXE 1

<b>Les objets en C++</b> . . . . .	357
<b>Les pointeurs</b> . . . . .	357
Manipulation d'adresses . . . . .	357
Exercice de manipulation des pointeurs . . . . .	359
<b>La mémoire en C++</b> . . . . .	360
La réservation mémoire . . . . .	361
Exemple . . . . .	361
Conclusion . . . . .	362

## ANNEXE 2

<b>La programmation objet</b> . . . . .	363
De l'analyse à la programmation objet . . . . .	363
Notion de machine virtuelle . . . . .	365
<b>Écrire des programmes</b> . . . . .	366
Concevoir . . . . .	366
Programmer . . . . .	366
Les notions à connaître . . . . .	368
Les erreurs à éviter . . . . .	369

## ANNEXE 3

<b>Liste des instructions du langage algorithmique</b> . . . . .	371
--	-----

## ANNEXE 4

<b>Notions de base d'UML</b> . . . . .	373
--	-----

<b>INDEX</b> . . . . .	377
------------------------	-----