

Avant-propos

Voici la deuxième édition de ce manuel d'introduction à la programmation orientée objet. À la demande de lecteurs, vous y découvrirez de nouveaux exercices et de nombreux exemples d'application dans trois langages : Java, C++ et Visual Basic.

L'objectif de cet ouvrage est d'apprendre à programmer dans des langages informatiques de haut niveau. Ne vous attendez donc pas à des programmes compliqués ou à des méthodes miracles pour créer en quelques clics une application parfaite. L'apprentissage de la programmation demande tout d'abord de maîtriser un certain nombre de concepts de base assez simples. Vous en étudierez ensuite d'autres un peu plus compliqués. En abordant ce livre sans connaissances particulières, vous saurez concevoir, à la fin de sa lecture, des programmes assez complexes dans plusieurs langages : vous développerez, par exemple, un logiciel de jeu de Puissance 4 que vous aurez plus de mal à battre qu'à programmer.

Le langage algorithmique

Un langage pour réfléchir sur papier

Ce livre s'articule autour d'un langage algorithmique inspiré du langage Java (qui est le langage d'application de beaucoup d'étudiants). Ce langage algorithmique possède deux avantages par rapport aux langages de programmation classiques (Java, C++, Visual Basic...).

- Il oblige le programmeur à travailler sur papier : la réflexion avec un crayon est en effet une étape nécessaire.
- Il permet de se libérer des contraintes liées à l'implémentation d'un langage sur ordinateur. Les notions algorithmiques sont introduites de manière simple, sans avoir à se préoccuper des particularités d'un langage.

Un langage transposable

Le langage algorithmique a un avantage didactique et pédagogique indéniable : il permet à l'étudiant de se détacher du programme pour se focaliser sur la réflexion. Cependant, si vous préférez apprendre l'algorithmique objet par le biais d'un langage exécutable sur ordinateur comme Java, C++ ou Visual Basic, ce manuel vous sera aussi utile puisque ce langage algorithmique peut être facilement

transposé dans n'importe quel langage objet existant. Le chapitre 12 présente ainsi des exemples d'applications des notions théoriques dans trois langages : Java, C++ et Visual Basic. L'annexe 3 donne d'ailleurs la correspondance entre le langage algorithmique et les langages de programmation Java, C++ et Visual Basic.

Structure de l'ouvrage

Le livre vous donne les outils pour bien concevoir un algorithme et un programme. Pour cela, chaque chapitre se clôt par une série d'exercices dont les solutions figurent dans la dernière partie de l'ouvrage. Les solutions en Java, C++ et Visual Basic de ces exercices sont accessibles depuis le site www.editions-eyrolles.com.

La première partie introduit longuement les variables et leur utilisation, les conditionnelles, les boucles et les fonctions.

La deuxième partie est consacrée aux objets : leur utilisation et leur écriture.

L'écriture de nouvelles classes et l'étude d'algorithmes essentiels seront abordées à travers des structures de données classiques, présentées dans la troisième partie.

Enfin, pour appliquer les connaissances concrètes acquises, ce livre se termine par la conception et l'écriture d'un jeu de Puissance 4 permettant de jouer contre l'ordinateur. Cette dernière partie contient également les solutions des exercices, ainsi que des exemples d'applications en Java, C++ et Visual Basic.

À qui s'adresse cet ouvrage ?

Cet ouvrage est destiné à tous ceux qui souhaitent s'initier à la programmation : il utilise une méthode qui favorise la réflexion et la conception.

Il s'adresse tout particulièrement aux étudiants de premier cycle en informatique qui pourront appréhender les notions et les méthodes de programmation qu'ils aborderont dans leurs études.

Enfin, les programmeurs désireux de découvrir l'approche orientée objet se concentreront sur les parties 2 à 4 pour comprendre toute la puissance de la conception et de la programmation objet.