

Disseny i programació orientada a objectes

Joan Arnedo Moreno
Daniel Riera i Terrén (coordinadors)

PID_00160182

Material docent de la UOC



Universitat Oberta
de Catalunya

www.uoc.edu

Joan Arnedo Moreno

Enginyer Informàtic per la Universitat Politècnica de Catalunya. Professor propi dels Estudis d'Informàtica, Multimèdia i Telecomunicació de la UOC. Professor associat de les assignatures de programació a l'Escola Tècnica Superior d'Enginyers de Telecomunicacions de Barcelona.

Daniel Riera i Terrén

Enginyer informàtic, màster en Informàtica industrial: tècniques avançades de la producció, i doctor per la Universitat Autònoma de Barcelona. Professor propi i director del Programa d'Enginyeria Informàtica als Estudis d'Informàtica, Multimèdia i Telecomunicació de la UOC.

Jordi Bríñquez Jiménez

Enginyer informàtic per la Universitat Politècnica de Catalunya i col·laborador de la UOC des d'octubre de 2003. Consultor de Programació Orientada a objectes, Laboratori de Java i assignatures de postgrau.

Elena García Barriocanal

Enginyer en Informàtica per la Universitat Pontifícia de Salamanca. Professora del departament de Ciències de la Computació de la Universitat d'Alcalà.

Miquel Àngel Piera i Eroles

Llicenciat en Informàtica per la Universitat Autònoma de Barcelona. Master en Control Engineering en l'UMIST (Regne Unit). Doctor en Informàtica per la UAB. Professor titular de la UAB.

Juan José Ramos González

Llicenciat en Informàtica per la Universitat Autònoma de Barcelona. Master en Informàtica Industrial: Tècniques Avançades de la Producció. Doctor en Informàtica per la UAB. Professor titular de l'Escola Universitària de la UAB.

Miguel Ángel Sicilia Urbán

Enginyer en Informàtica per la Universitat Pontifícia de Salamanca. Professor del departament d'Informàtica de la Universitat Carlos III.

Albert Tort Pugibet

Enginyer d'Informàtica i màster en Computació per la Universitat Politècnica de Catalunya (UPC). Membre del Grup de Recerca Modelització i Processament d'Informació (MPI) i del Departament d'Enginyeria de Serveis i Sistemes d'Informació de la UPC.

Primera edició: setembre 2010

© Jordi Bríñquez Jiménez, Elena García Barriocanal, Miquel Àngel Piera i Eroles, Juan José Ramos González, Miguel Ángel Sicilia Urbán, Albert Tort Pugibet

Tots els drets reservats

© d'aquesta edició, FUOC, 2010

Av. Tibidabo, 39-43, 08035 Barcelona

Disseny: Manel Andreu

Material realitzat per Eureka Media, SL

ISBN: 978-84-693-4313-5

Dipòsit legal: B-27.193-2010

Cap part d'aquesta publicació, incloent-hi el disseny general i de la coberta, no pot ser copiada, reproduïda, emmagatzemada o transmesa de cap manera ni per cap mitjà, tant si és elèctric, com químic, mecànic, òptic, de gravació, de fotocòpia, o per altres mètodes, sense l'autorització prèvia per escrit dels titulars del copyright.

Introducció

En aquesta assignatura s'introdueix el paradigma de la programació orientada a objectes. Aquest es considera una evolució del paradigma de programació estructurada que permet la modelització d'objectes del món real, amb les seves característiques i propietats internes.

La màxima de Niclaus Wirth “algorismes + estructures de dades = programes” esdevé clau per a descriure els components bàsics dels llenguatges de programació orientada a objectes: els objectes, conceptes que contenen tant les dades (estructures de dades) com els mètodes que les manipulen (algorismes). A més, l'organització dels programes en objectes permet assolir certs avantatges a l'hora de desenvolupar aplicacions complexes.

Dins d'aquesta assignatura farem servir el llenguatge de programació Java, el qual ha anat evolucionant al ritme dels canvis proposats pels desenvolupadors i ha acabat essent una eina molt potent i àmpliament utilitzada. Aquest llenguatge va ser concebut per a simplificar algunes de les propietats més difícils d'implementar del paradigma de la programació orientada a objectes i, a més, és independent de la plataforma d'execució, tot i que això tregui un cert control al programador.

El material docent d'aquesta assignatura està estructurat en quatre grans blocs. Un primer bloc introdueix el paradigma de la programació orientada a objectes seguint una aproximació clàssica: a partir de la presentació de les deficiències del paradigma de la programació estructurada, es defineixen els conceptes de classe i objecte, peces bàsiques de la programació orientada a objectes. Igualment, es presenten l'abstracció i la classificació, que expliquen possibles relacions entre aquests.

El segon bloc comença amb un mòdul d'introducció de conceptes de programació avançats, necessaris per a la comprensió i programació d'objectes i les relacions entre aquests. A aquest, el segueix un mòdul dedicat als conceptes d'herència i de polimorfisme, aportacions fonamentals de la programació orientada a objectes. Finalment, aquest bloc també descriu els diagrames UML com a eina bàsica per a la representació de manera simbòlica del conjunt de classes d'un problema donat i les seves relacions.

El tercer bloc presenta un exemple pràctic fet des de l'especificació fins a la codificació aplicant els conceptes de disseny que s'han après al llarg del curs.

Finalment, el quart bloc introdueix el llenguatge Java com a eina bàsica per als exercicis pràctics de programació. Aquest bloc descriu tant les característiques

que permeten fer servir Java com a llenguatge de programació estructurada, com les que estan directament lligades als conceptes bàsics de la programació orientada a objectes.

Objectius

Els materials didàctics d'aquesta assignatura aportaran a l'estudiant els coneixements necessaris per a tenir una visió clara del paradigma de la programació orientada a objectes i li permetran assolir els objectius següents:

- 1.** Conèixer els conceptes bàsics de la programació orientada a objectes seguint un fil argumental: classes i objectes, abstracció i classificació, herència, sobrecàrrega i polimorfisme.
- 2.** Resoldre problemes de disseny usant una descomposició basada en l'abstracció i la classificació: les associacions entre objectes i classes.
- 3.** Identificar i descriure les classes que ens permetin representar i modelar els elements d'un problema i les relacions entre els mateixos a través de diagrames UML.
- 4.** Aprendre a programar en Java utilitzant els recursos propis de la programació orientada a objectes.
- 5.** Adquirir destresa en la utilització de les eines de desenvolupament en Java.

Continguts

Mòdul didàctic 0

Contextualització

Albert Tort Pugibet

1. Introducció als paradigmes de programació
2. Cicle de vida del desenvolupament de programari
3. Arquitectura en tres capes

Mòdul didàctic 1

Introducció a l'orientació a objectes

Jordi Brínquez Jiménez, Elena García Barriocanal

1. Els inconvenients de la programació clàssica
2. L'orientació a objectes
3. Llenguatges de programació orientada a objectes

Mòdul didàctic 2

Classes i objectes

Jordi Brínquez Jiménez, Miquel Àngel Piera i Eroles, Juan José Ramos González

1. Criteris de qualitat
2. El principi de modularitat
3. Classes i objectes

Mòdul didàctic 3

Abstracció i classificació

Jordi Brínquez Jiménez, Miguel Ángel Sicilia Urbán

1. Complexitat i abstracció
2. Els principis de la classificació
3. Relacions entre classes
4. Tècnica simple per a identificar classes
5. Tipus de classes

Mòdul didàctic 4

Estructures d'objectes

Jordi Brínquez Jiménez

1. Instanciació
2. Recorreguts
3. Delegació

Mòdul didàctic 5

Herència i polimorfisme

Jordi Brínquez Jiménez

1. Definició d'herència
2. Consideracions sobre l'herència
3. Errors de l'herència
4. Polimorfisme
5. Implementació de l'herència i el polimorfisme amb Java

Mòdul didàctic 6

Un exemple pràctic

Jordi Brínquez Jiménez

1. Enunciat
2. Resolució

Mòdul didàctic 7

El llenguatge de programació Java

Jordi Brínquez Jiménez

1. Java com a llenguatge de programació estructurada
2. Java com a llenguatge de programació orientada a objectes

Bibliografia

Booch, G.; Rumbaugh, J.; Jacobson, I. (1999). *El lenguaje de modelado unificado UML*. Madrid: Addison-Wesley Iberoamericana.

Eckel, B. (2000, 2003). *Thinking in C++* (2a. ed., vol. 1 i 2). Nova Jersey: Prentice Hall.

Horstmann, C. S. (1997). *Practical object oriented development in C++ and Java*. Hoboken: John Wiley & Sons.

Joyanes, L. (1998) *Programación orientada a objetos*. Madrid: McGraw-Hill.

Kernighan, B. W.; Ritchie, D. M. (1988). *The C programming language* (2a. ed.). Nova Jersey: Prentice Hall

Meyer, B. (1999). *Construcción de software orientado a objetos*. Madrid: Prentice Hall.

Rumbaugh, J.; Blaha, M.; Premerlani, W.; Eddy, F.; Lorensen, W. (1996). *Modelado y diseño orientado a objetos*. Madrid: Prentice Hall.

Stroustrup, B. (1997). *The C++ programming language* (3a. ed.). Massachusetts: Addison Wesley.

Webs

C++ Reference: <http://www.cppreference.com/>

C++ Standard Library: <http://www.cplusplus.com/ref/>

C++ FAQ Lite: <http://www.parashift.com/c++-faq-lite/>

Wikipedia: <http://www.wikipedia.org>