



Pla Docent de l'Assignatura

Guia Docent Programació d'Activitats

Curs acadèmic:	2011-12	Trimestre: 1r
Nom de l'assignatura:	Programació Orientada a Objectes	
Codi assignatura:	21414, 21721 i 21603	
Estudis:	Grau en Enginyeria en Informàtica / Grau en Enginyeria Telemàtica / Grau en Enginyeria en Sistemes Audiovisuals	
Nombre de crèdits ECTS:	4	
Nombre total d'hores de dedicació:	100	
Temporalització:	Curs: 2n curs Tipus: Trimestral Període: 1r trimestre	
Professorat:	Daniel Soto, Anders Jonsson, Oriol Galimany, Ayman Moghnieh	
Grup:	1,2	

Guia Docent

1. Dades descriptives de l'assignatura

- **Curs acadèmic:** 2011-2012
- **Nom de l'assignatura:** Programació Orientada a Objectes **Codi:** 21414, 21721 i 21603
- **Tipus d'assignatura:** Troncal
- **Titulació / Estudis:** Grau en Enginyeria en Informàtica, Grau en Enginyeria Telemàtica i Grau en Enginyeria en Sistemes Audiovisuals
- **Nombre de crèdits:** 5 **Nombre d'ECTS:** 4
- **Nombre total d'hores de dedicació a l'assignatura:** 100
- **Temporalització:**
 - Curs: 2n
 - Tipus: Trimestral
 - Període: 1r trimestre
- **Coordinació:** Daniel Soto i Anders Jonsson
- **Departament:** Departament de Tecnologies de la Informació i les Comunicacions
- **Professorat:** Daniel Soto, Anders Jonsson, Oriol Galimany, Ayman Moghnieh
- **Grup:** 1, 2
- **Llengua de docència:** Català/Castellà
- **Edifici on s'imparteix l'assignatura:** Roc Boronat

2. Presentació de l'assignatura

Programació Orientada a Objectes és una assignatura dins l'àrea de Llenguatges de Programació, fortament relacionada amb Fonaments de Programació i Estructura de Dades i Algorismes (aquesta darrera pertany només al grau d'Enginyeria en Informàtica). El punt de partida és la programació estructurada i l'algorísmica bàsica (coneixements adquirides en l'assignatura de Fonaments de Programació), i té com a objectiu introduir a l'alumne en el disseny i desenvolupament de programes d'alt nivell mitjançant el concepte d'objectes.

Aquesta assignatura precisa necessàriament d'un coneixement bàsic dels conceptes de programació, doncs el paradigma de la programació orientada a objectes està principalment enfocat a millorar el disseny dels propis programes. D'aquesta manera, s'introdueix a l'alumne en la creació de programes des d'un punt de vista més abstracte que la simple algorísmica bàsica, promovent un bon disseny enfocat a resoldre els aspectes més rellevants del problema, mentre allò més específic queda ocult. Així, a diferència dels cursos orientats a la programació procedural, a on la sintaxis i semàntica del llenguatge són primordials, aquest curs s'orienta cap al disseny i estructura dels programes en el seu conjunt. No obstant això, també s'introdueix i s'ensenya el llenguatge Java per implementar els programes dissenyats.

L'objectiu principal de l'assignatura és estudiar els principis necessaris per a modelar un programa utilitzant diferents components, els quals s'han de combinar per tal d'aconseguir resoldre un problema concret. Addicionalment això comporta ser capaç de reutilitzar components ja existents per a la construcció de nous programes. Igualment s'estudien diferents tècniques per a la definició i construcció d'aquestes components.

L'organització de l'assignatura segueix la següent estructura: Al principi del curs s'estudien els conceptes i principis fonamentals de la programació orientada a objectes, com són: l'abstracció, l'encapsulació i la reutilització. Es defineix el concepte d'objecte com a eina que modela i descriu una entitat capaç de realitzar una sèrie de tasques concretes. També es mostra com s'aplica el concepte de la delegació de tasques per a coordinar el funcionament dels diferents objectes dintre d'un programa. En una segona part del curs, utilitzant els coneixements bàsics anteriors, s'introdueixen conceptes més avançats com ara: l'herència, el polimorfisme, la sobrecàrrega i les interfícies; estudiant especialment com aplicar-los i utilitzar-los en la construcció de programes orientats a objectes. En la darrera part del curs, s'aprèn com aplicar tots els conceptes en el seu conjunt per a resoldre problemes complexos. Això es desenvolupa mitjançant l'estudi del modelat d'objectes i les seves relacions, així com la reutilització de codi, i també l'estudi de problemes concrets resolts utilitzant objectes.

Les diferents activitats realitzades durant el curs es divideixen en:

- Sessions de teoria: a on s'introdueixen i s'expliquen els diferents conceptes i principis de la programació orientada a objectes.
- Sessions de pràctica: a on els alumnes, organitzats en grups, tenen que treballar per a implementar programes escrits en llenguatge Java que resolguin diferents problemes, tot aplicant les tècniques exposades en les classes de teoria.
- Sessions de seminaris: a on els alumnes, de forma individual, tenen que dissenyar i modelar, utilitzant objectes, una solució als problemes particulars presentats, la qual més tard tindran que implementar en les sessions de pràctiques.

Al final de l'assignatura els alumnes estaran capacitats per a:

- Implementar programes utilitzant un llenguatge de programació d'alt nivell orientat a objectes (particularment Java).
- Entendre l'estructura d'un programa existent organitzat amb objectes.
- Modelar una solució basada en objectes que resolgui un problema particular.
- Reutilitzar objectes existents i utilitzar i/o implementar interfícies.

3. Prerequisits per al seguiment de l'itinerari formatiu

Aquesta assignatura pressuposa que els alumnes coneixen els fonaments de l'algorísmica bàsica i la programació estructurada, i que per tant són capaços d'escriure programes i resoldre problemes utilitzant llenguatges imperatius d'alt nivell (competències adquirides en l'assignatura de Fonaments de Programació).

4. Competències a assolir en l'assignatura

Competències generals	Competències específiques
Instrumentals 1. Capacitat d'anàlisi del problema 2. Capacitat d'abstracció Interpersonals 3. Comprensió de solucions ja existents o fetes per d'altres Sistèmiques 4. Creació de solucions fortament estructurades 5. Utilització de documentació en el desenvolupament 6. Reutilització de solucions Altres 7. Treball col·laboratiu en el desenvolupament de software	 1. Comprensió del rol de l'objecte en l'organització d'un programa 2. Capacitat de dissenyar solucions utilitzant objectes 3. Domini de les tècniques bàsiques de la programació orientada a objectes (herència, abstracció, polimorfisme) 4. Reutilització de codi utilitzant classes existents 5. Utilització del concepte d'interfície per a relacionar objectes 6. Poder entendre i escriure codi en Java

5. Objectius d'aprenentatge

L'objectiu principal de l'assignatura és l'adquisició de les competències necessàries per a poder treballar en el desenvolupament de software (disseny i implementació) utilitzant el paradigma de la programació orientada a objectes. Això inclou d'altres objectius parcials com són adquirir la capacitat d'abstracció necessària per a encapsular solucions parcials dintre d'un objecte, relacionar correctament diferents objectes dintre d'un programa per donar una solució ben estructurada a un problema concret, i reutilitzar elements existents dintre dels propis programes. De forma paral·lela, també s'inclou l'objectiu del domini elemental d'un llenguatge de programació d'alt nivell orientat a objectes, com ara el llenguatge Java. Finalment també es defineix l'objectiu d'entendre i escriure diagrames que reflecteixin l'estructura d'un programa orientat a objectes.

De forma concreta, al final de l'assignatura s'espera que l'alumne pugui:

- a) Entendre un programa escrit amb un llenguatge orientat a objectes.
- b) Implementar un programa escrit amb un llenguatge orientat a objectes.
- c) Comprendre l'estructura d'un programa existent organitzat amb objectes.
- d) Construir una solució a un determinat problema utilitzant objectes.
- e) Reutilitzar objectes existents en els propis programes.
- f) Definir i implementar interfícies per a connectar diferents objectes dintre d'un programa.
- g) Escriure i llegir codi en llenguatge Java.

6. Avaluació

6.1. Criteris generals d'avaluació

L'avaluació de l'assignatura es basa en mesurar l'aprenentatge de l'alumne respecte als continguts presentats. Les eines utilitzades per a la realització d'aquesta mesura seran les següents:

- a) Seguiment individualitzat continu en les classes pràctiques i seminaris.
- b) Avaluació per mitjà d'un examen teòric individual.
- c) Avaluació conjunta de les pràctiques.
- d) Avaluació conjunta dels exercicis realitzats als seminaris.

Cada tasca serà avaluada amb una nota numèrica amb un valor entre 0 i 10. La següent taula presenta una visió completa de l'avaluació a realitzar:

Tipus d'avaluació	Incidència qualificació		Agent responsable de l'avaluació			Agrupació		%
	Obligatòria	Optativa	Docent	Autoavaluació	Coavaluació	individual	grups	
Continuada	Exercicis seminaris		Professor seminari			X		10%
Formativa	Pràctiques		Professor pràctiques				X	40%
Final	Examen final		Professor teoria			X		50%

Per tal d'aprovar l'assignatura serà necessari superar en un 50% cada grup d'avaluació, essent condició necessària treure una nota igual o superior a 5,0 (sobre 10) tant del bloc de pràctiques com de l'examen final.

En cas de no superar correctament alguna de les avaluacions, l'alumne tindrà el dret de mantenir la mateixa nota en els blocs superats de cara a la segona convocatòria.

Per a la segona convocatòria de l'assignatura només s'avaluaran les pràctiques i l'examen extraordinari de setembre. El contingut de les pràctiques serà similar, però no idèntic, al lliurat durant el curs; i s'haurà de fer en un únic lliurament segons les indicacions del professor.

7. Continguts

Bloc de contingut 1. Introducció: el concepte d'objecte

Conceptes	Procediments
1. Definició intuïtiva d'un objecte 2. Resolució de problemes amb objectes 3. Delegació de tasques 4. Com relacionar objectes entre sí 5. L'ús d'objectes en la programació	

Bloc de contingut 2. El paradigma de la programació orientada a objectes

Conceptes	Procediments
1. Classe 2. Instància 3. Jerarquia de classes 4. Abstracció 5. Encapsulació	Definició de les classes, els seus atributs i mètodes

Bloc de contingut 3. Tècniques fonamentals: herència i polimorfisme

Conceptes	Procediments
1. Herència 2. Polimorfisme 3. Sobrecarrega 4. Reescriptura	Heretar membres d'una superclasse Modificació del comportament i l'estructura d'una classe

Bloc de contingut 4. Conceptes avançats: interfícies i classes abstractes

Conceptes	Procediments
1. Mètodes abstractes 2. Classes abstractes 3. Interfícies 4. Connexió d'objecte mitjançant interfícies	Disseny de classes amb mètodes abstractes i implementació en les seves subclasses Desenvolupament d'interfícies per a l'organització dels programes Implementació d'interfícies

Bloc de contingut 5. Modelat d'objectes i relacions entre objectes

Conceptes	Procediments
1. Disseny de classes 2. Tècniques de modelat d'objectes 3. Relacions entre objectes 4. Combinació d'objectes para resoldre problemes concrets	Disseny de solucions a problemes utilitzant diferents classes Implementació de programes utilitzant una solució modelada amb objectes

Bloc de contingut 6. Reutilització i estudi de problemes resolts amb objectes

Conceptes	Procediments
1. Reutilització de classes existents 2. Implementació d'interfícies 3. Estudi de casos particulars de problemes	Utilització de classes complexes existents en els propis programes Reutilització de classes per definir noves Utilització i implementació d'interfícies definides un API concret

Bloc de contingut 7. Eines avançades i programació visual

Conceptes	Procediments
1. Eines de modelar 2. Programació visual 3. El futur de la POO	

8. Metodologia

8.1. Enfocament metodològic de l'assignatura

En aquesta secció es presenta de forma resumida la metodologia d'aprenentatge que s'utilitzarà en l'assignatura. Aquesta guia és especialment útil per l'alumne doncs podrà conèixer les activitats que estan planificades per a l'aprenentatge, i organitzar d'aquesta forma els esforços que haurà de dedicar-li. Totes les activitats estan englobades dintre dels següents àmbits:

- Activitats presencials (les realitzades dintre de l'aula)
- Activitats dirigides (les executades fora de l'aula)
- Activitats autònomes (les que realitza l'alumne de forma individual)
- Activitats d'aprenentatge en equip (les que es porten a terme amb d'altres alumnes i sense supervisió directa)

8.1. Sessions de teoria

En les sessions de teoria l'alumne hi participará de forma presencial rebent els coneixements teòrics explicats pel professor. Aquestes classes es complementen amb els apunts de teoria que l'alumne podrà revisar en activitats autònomes.

En les sessions de teoria hi seran presents tots els alumnes d'un determinat grup.

8.2. Sessions de pràctiques

Les sessions de pràctiques s'organitzen en dos tipus d'activitats:

En la primera activitat, de tipus presencial, els alumnes reben les explicacions del professor respecte als objectius específics que es tractaran en la sessió a realitzar, així com els coneixements tècnics necessaris per a poder dur a terme la tasca proposada.

En la segona activitat, d'aprenentatge en equip, els alumnes posen en pràctica tots els coneixements adquirits per a realitzar els objectius proposats. Aquesta activitat continuarà fora de l'aula fins aconseguir l'objectiu necessari.

En les sessions de pràctiques els alumnes s'organitzen en grups de dues persones per a realització de les diferents activitats.

8.3. Sessions de seminaris

En les sessions de seminaris es plantegen diferents exercicis encaminats a dirigir a l'alumne de forma didàctica en la resolució correcta de diferents exercicis. Aquesta activitat serà sempre de tipus presencial i autònoma, doncs l'objectiu és ajudar a l'alumne a raonar amb un cert grau d'abstracció per a dissenyar una determinada solució.

En cada seminari hi participará només un grup reduït d'alumnes (subgrup) segons el calendari establert de l'assignatura. Al final de cada sessió els alumnes hauran de

lliurar els exercicis realitzats per tal d'avaluar el seu progrés i poder rebre comentaris del professor en la següent sessió.

8.4. Activitats d'aprenentatge autònom

Per a la realització de certes activitats, com ara la implementació d'un determinat programa o la preparació d'una sessió de seminaris, s'espera que l'alumne sigui responsable en l'adquisició dels coneixements i habilitats complementaris necessaris per a poder completar l'activitat. Especialment s'inclouen en aquest conjunt les consultes de documentació tècnica, bibliografia i d'altres guies per a la codificació dels programes. Igualment s'inclou la revisió prèvia de material complementari recomanat pel professor per a la realització de determinats seminaris.

8.5. Activitats d'aprenentatge en grup

Les activitats d'aprenentatge en grup són principalment les relacionades amb les sessions de pràctiques. No obstant això, les activitats en grup seran només presencials durant aquestes sessions pràctiques, i no dirigides durant la resta del temps necessari per a completar la seva realització.

9. Fonts d'informació i recursos didàctics

9.1. Fonts d'informació per a l'aprenentatge. Bibliografia bàsica (suport paper i electrònic)

Teoria:

Meyer: Object Oriented Software Construction (disponible en castellà "Construcción de Software Orientado a Objetos")

Horstmann: Object Oriented Design and Patterns

Pràctiques:

Eckel: Thinking in Java (disponible en castellà "Piensa en Java" i en versió online)

Arnold, Gosling, Holmes: Java Programming Language (disponible en castellà "El lenguaje de programación Java")

9.2. Fonts d'informació per a l'aprenentatge. Bibliografia complementària (suport paper i electrònic)

Netbeans: http://java.sun.com/developer/onlineTraining/tools/netbeans_part1/

9.3. Recursos didàctics. Material docent de l'assignatura

En l'aula digital Campus Global (Moodle) de l'assignatura es publicarà el següent material docent:

- Apuntes de les classes de teoria
- Enunciats i apunts de les classes pràctiques
- Enunciats de les sessions de seminaris
- Enllaços a informació complementària

Programació d'Activitats

Grup 1

Setmana				Teoria	Seminari	Pràctiques	Personal	Total Hores
	Dilluns 18:30-20:30	Dimecres 16:30-18:30	Dijous 14:30-16:30					
1			Teoria T1 (1) Introducció: el concepte d'objecte	2				2
2	Pràctica P11,P13 (1) Introducció a l'IDE (Netbeans) Compilació de programes en Java	Pràctica P12 (1) Introducció a l'IDE (Netbeans) Compilació de programes en Java	Teoria T1 (2) El paradigma de la programació orientada a objectes	2		2	4	8
3	Seminaris S111,S112, S131 (1) Exercicis de Modelat	Seminaris S121,S122 (1) Exercicis de Modelat	Teoria T1 (3) Conceptes fonamentals: la Herència	2	2		6	10
4	Pràctica P11,P13 (2) Definir una classe a Java		Pràctica P12 (2) Definir una classe a Java			2	6	8
5	Seminaris S111,S112, S131 (2) Exercicis de Modelat	Seminaris S121,S122 (2) Exercicis de Modelat	Teoria T1 (4) Conceptes fonamentals: el Polimorfisme	2	2		6	10
6	Pràctica P11,P13 (3) Herència: implementar classes que hereten unes d'altres	Pràctica P12 (3) Herència: implementar classes que hereten unes d'altres	Teoria T1 (5) Conceptes avançats: interfícies i classes abstractes	2		2	8	12
7		Seminaris S111,S112, S131 (3) Exercicis de Modelat	Seminaris S121,S122 (3) Exercicis de Modelat		2		8	10

8	Pràctica P11,P13 (4) Reutilitzar codi existent mitjançant l'API	Pràctica P12 (4) Reutilitzar codi existent mitjançant l'API	Teoria T1 (6) Modelat i relacions entre objectes	2		2	8	12
	Seminaris S111,S112, S131 (4) Exercicis de Modelat	Seminaris S121,S122 (4) Exercicis de Modelat	Teoria T1 (7) Reutilització de codi	2	2		6	10
	Pràctica P11,P13 (5) Entrada/Sortida amb objectes	Pràctica P12 (5) Entrada/Sortida amb objectes	Teoria T1 (8) Estudi de casos particulars resolts amb objectes	2		2	6	10
			Teoria T1 (9) Eines avançades i programació visual	2			6	8
				18	8	10	64	100

Grup 2

Setmana				Teoria	Seminari	Pràctiques	Personal	Total Hores
	Dimarts 16:30-18:30	Dijous 18:30-20:30	Divendres 14:30-16:30					
1			Teoria T2 (1) Introducció: el concepte d'objecte	2				2
2	Pràctica P21 (1) Introducció a l'IDE (Netbeans) Compilació de programes en Java	Pràctica P22,P23 (1) Introducció a l'IDE (Netbeans) Compilació de programes en Java	Teoria T2 (2) El paradigma de la programació orientada a objectes	2		2	4	8
3	Seminaris S211,S212 (1) Exercicis de Modelat	Seminaris S221,S222, S231 (1) Exercicis de Modelat	Teoria T2 (3) Conceptes fonamentals: la Herència	2	2		6	10
4	Pràctica P21 (2) Definir una classe a Java	Pràctica P22,P23 (2) Definir una classe a Java				2	6	8

5	Seminaris S211,S212 (2) Exercicis de Modelat	Seminaris S221,S222, S231 (2) Exercicis de Modelat	Teoria T2 (4) Conceptes fonamentals: el Polimorfisme	2	2		6	10
	Pràctica P21 (3) Herència: implementar classes que hereten unes d'altres	Pràctica P22,P23 (3) Herència: implementar classes que hereten unes d'altres	Teoria T2 (5) Conceptes avançats: interfícies i classes abstractes	2		2	8	12
7		Seminaris S211,S212 (3) Exercicis de Modelat	Seminaris S221, S222,S231 (3) Exercicis de Modelat		2		8	10
		Pràctica P21 (4) Reutilitzar codi existent mitjançant l'API	Pràctica P22,P23 (4) Reutilitzar codi existent mitjançant l'API	2		2	8	12
9	Seminaris S211,S212 (4) Exercicis de Modelat	Seminaris S221,S222, S231 (4) Exercicis de Modelat	Teoria T2 (7) Reutilització de codi	2	2		6	10
	Pràctica P21 (5) Entrada/Sortida amb objectes	Pràctica P22,P23 (5) Entrada/Sortida amb objectes	Teoria T2 (8) Estudi de casos particulars resolts amb objectes	2		2	6	10
11			Teoria T2 (9) Eines avançades i programació visual	2			6	8
				18	8	10	64	100