

Codi: 21307

Metodologia i Tecnologia de la Programació II

3r Semestre
Enginyeria Tècnica Informàtica
Escola Universitària de Sabadell (UAB)

[[Novetats](#)] [[Docència](#)] [[Objectius](#)] [[Temari](#)] [[Bibliografia](#)] [[Pràctiques](#)]
[[Avaluació](#)]

Novetats 2003-04

- 22/09/2003 - Dijous 2 d'Octubre comencen les classes

Docència

Professor	Part	Horaris	Consultes
Ramon Baldrich	Teoria	[horaris de teoria]	[horaris consultes]x [despatx]
Javier Jiménez	Problemes	Dijous de 10 a 11 hores Dijous de 16 a 17 hores	Dijous de 14 a 16 hores al despatx S246 o a l'aula C
	Pràctiques (Aula C)	G40, dimarts de 13 a 15 hores G50, dimarts de 15 a 17 hores G60, dimarts de 17 a 19 hores G70, dimarts de 19 a 21 hores G90, dijous de 17 a 19 hores	
Alicia Fornés	Pràctiques	G00, dilluns de 9 a 11	[horaris consultes]x


	(Aula C)	hores G10, dilluns d'11 a 13 hores G20, dilluns de 13 a 15 hores G30, dilluns de 15 a 17 hores G80, dijous de 11 a 13 hores	[despatx]
--	----------	---	-----------

Objectius

L'objectiu de l'assignatura és proveir a l'alumnat amb el coneixement necessari per tal d'abordar problemes de complexitat major, mitjançant tècniques que li permetin analitzar un programa (correcció, cost, ...), i dissenyar noves solucions a partir de la idea de recursivitat o de tècniques de disseny d'algorismes.

Temari

El número entre parèntesis és el nombre de sessions de classe.

1. PRESENTACIÓ DE L'ASSIGNATURA (1)
2. PROGRAMACIÓ EN C++ (5) 
 - Introducció.
 - Classes, objectes, constructors i destructors.
Mètodes interns i mètodes externs.
 - Derivació de classes. Accés a membres heretats.
Tipus d'accés a la classe base. Conversions i assignacions. Especialització. Composició vs herència. Herència múltiple.
 - Apuntadors i emmagatzematge dinàmic.
Aritmètica d'apuntadors. Operadors *new* i *delete*.
Apuntadors a objectes. Funcions: paràmetres per valor i per referència, retorns per referència.
Funcions virtuals.

- Sobrecàrrega de funcions i operadors. Funcions *Friend*.
 - *Templates*. Funcions genèriques. Sobrecàrrega de funcions genèriques. Classes genèriques. Herència de classes genèriques.
 - Conversions de tipus. Conversions explícites. Constructors i operadors de conversió. Operadors d'assignació-conversió.
 - Facilitats d'entrada / sortida.
3. RECURSIVITAT (4)
- Metodologia de disseny d'algorismes recursius. Prova de la correcció i l'acabament. Exemples: factorial i exponencial de nombres naturals.
 - Anàlisi d'algorismes recursius: càlcul de la complexitat. Equacions de recurrència. Exemples: Merge-sort, multiplicació de naturals.
 - Implementació dels algorismes recursius fent servir piles de recursió. Concepte de registre d'activació.
 - Tècniques de transformació recursiva-iterativa d'algorismes. Eliminació de la recursivitat final i no final. Recursivitat múltiple. Exemples.
4. COMPLEXITAT DELS ALGORISMES (2)
- Introducció. Criteris per seleccionar un algorisme. Factors que influeixen en el temps d'execució.
 - Exemple: càlcul del cost d'un algorisme d'ordenació per inserció.
 - Notació asimptòtica: Theta-notació, O-notació, omega-notació.
 - Càlcul del temps d'execució d'un programa. Suma i multiplicació en notació asimptòtica. Suggerències pràctiques pel càlcul de la complexitat.
 - Comparació entre les complexitats més usuals.
5. TÈCNiques DE DISSENY D'ALGORISMES (6)
- Introducció.
 - Tècnica incremental. Exemple: Ordenació per inserció.
 - *Divide and Conquer*. Esquema general de la tècnica. Exemples: Merge-sort, les torres de Hanoi. Equacions de recurrència associades.

- Mètodes de *Backtracking* i *Branch & Bound* .
Backtracking. Exemple: problema de les n reines.
Branch & Bound. Exemple: puzzle de 15 peces.
 - Programació dinàmica. Idea general del mètode.
Exemple: trobar el camí més curt entre ciutats.
 - Tècniques *Greedy* . Esquema general. Exemple:
problema de la motxilla.
6. MODULARITZACIÓ (1)
- Introducció. Avantatges de la modularització.
 - Conceptes fonamentals: mòdul, disseny modular, exportacions, importacions, privacitat.
 - Creadors i usuaris. Qualificació d'importacions.
 - Criteris de descomposició modular.
 - Implementació de mòduls en C.
 - Exemple concret: mòdul de cadenes de caràcters.
7. ESPECIFICACIÓ I DERIVACIÓ D'ALGORISMES ITERATIUS [I] (2)
- Concepte d'estat i d'espai d'estats d'un programa.
 - Especificació d'un algorisme. Elements d'una especificació. Precondicions, postcondicions. Regles de conseqüència d'una especificació.
8. ESPECIFICACIÓ ALGEBRAICA DE TIPUS DE DADES (3)
- Conceptes de signatura, especificació algebraica, àlgebra de termes i reescriptura. Mètode per donar una especificació formal.
 - Exemples d'especificacions algebraiques: Naturals. Llistes. Piles. Cues. Arbres.
 - Utilitat de la programació funcional en l'especificació de tipus de dades i programes.
-

Bibliografia

- A. Aho, J. Hopcroft, J. Ullman. **Estructuras de datos y algoritmos**, *Addison Wesley*, 1987.
- G. Brassard, P. Bratley. **Fundamentos de algoritmia**, *Prentice Hall*, 1997.
- J. Arsac. **Las bases de la programación**, *Ed. Omega*, 1986.

- S. Baase. **Computer Algorithms. Introduction to Design and Analysis**, *Addison-Wesley Publishing Company*, 2nd edition.
 - J.L. Balcázar. **Programación metódica**, *Mc Graw Hill*, 1993.
 - J.Castro et. al. **Curs de programació**, *Mc Graw Hill*, 1ª edició, 1992.
 - T. Cormen, C. Leiserson, R. Rivest. **Introduction to algorithms**, *Mc Graw Hill*, 1990.
 - F. Escudero, J.M. Garrell. **Fonaments de programació**, *Ed. Bruño / EUETT*, 1ª edició, 1993.
 - X. Franch Gutiérrez. **Estructures de dades. Especificació, disseny i implementació**, *Edicions UPC*, 1993.
 - S. Hekmatpour. **C++ A Guide for C Programmers**, *Prentice Hall*, 1990.
 - E. Hernández Orallo, J. Hernández Orallo. **Programación en C++**, *Editorial Paraninfo*, 1993.
 - E. Horowitz, S. Sahni. **Fundamentals of computer algorithms**, *Computer Science Press*, 1989.
 - L. Joyanes Aguilar. **Fundamentos de programación (Algoritmos y Estructuras de datos)**, *Mc Graw Hill*, 1ª edició, 1988.
 - L. Kronsjö. **Algorithms: Their Complexity and Their Efficiency**, *Ed. John Wiley & Sons*.
 - B. Stroustrup. **The C++ Programming Language**, *Addison Wesley*, 1987.
 - N. Wirth. **Algoritmos y Estructuras de Datos**, *Prentice Hall*, 1986.
-

Pràctiques

Hi ha deu grups de pràctiques presencials, més un, de no presencial. Tots els alumnes s'han de registrar en un d'aquests grups a través del servei de gestió de grups de pràctiques [GGP](#). La inscripció al grup no presencial es reserva pels alumnes que justifiquin incompatibilitat d'horaris per motius de treball. Als grups presencials no ha d'haver-hi inscripcions sense emparellar, sempre que sigui possible.

	Dilluns	Dimarts	Dimecres	Dijous	Divendres
09:00 a 10:00					
10:00 a 11:00	00				
11:00 a 12:00					
12:00 a 13:00	10			80	
13:00 a 14:00					
14:00 a 15:00	20	40			
15:00 a 16:00					
16:00 a 17:00	30	50			
17:00 a 18:00					
18:00 a 19:00		60		90	
19:00 a 20:00					
20:00 a 21:00		70			

Les pràctiques es realitzen al llarg de deu sessions. L'assistència a les entregues és obligatòria. El calendari és el següent:

Sessió	Setmana	Breu comentari	Fitxers auxiliars
---	06/10/03		
1	13/10/03	Introducció a C++	
2	20/10/03	Introducció a C++	
3	27/10/03	SESSIÓ D'ENTREGA	
4	03/11/03	Recursivitat	
5	10/11/03	Recursivitat	
6	17/11/03	SESSIÓ D'ENTREGA	
7	24/11/03	Algorísmica	

8	01/12/03	Algorísmica	
---	08/12/03		
9	15/12/03	Algorísmica	
---	22/12/03		
---	29/12/03		
---	05/01/04		
10	12/01/04	SESSIÓ D'ENTREGA	

No deixeu de llegir la [normativa](#) de l'assignatura.

Avaluació

Per aprovar l'assignatura cal aprovar per separat la teoria (examen) i les pràctiques.

$$\text{Nota final} = 0.7 * \text{Nota teoria} + 0.3 * \text{Nota pràctiques}$$

A més hi ha la [normativa interna](#) de la Unitat de Processament d'Imatges i Intel.ligència Artificial del Departament d'Informàtica de la UAB.

*Departament d'Informàtica
Universitat Autònoma de Barcelona
08193 Cerdanyola (Barcelona)*