

METODOLOGIA I TECNOLOGIA DE LA PROGRAMACIÓ I

Codi	Tipus	Curs/semestre	Crèdits ECTS
26986	Anual	1r curs/1r i 2n semestre	12

Professors

<i>Nom</i>	<i>Grups</i>	<i>Dpt.</i>	<i>Despatx</i>	<i>Direcció e-mail</i>	<i>Telèfon</i>
Pere Guitart	10 i 20	dEIC	S/260	pguitart@deic.uab.cat	937287757
Ramon Musach	50 i lab 2	dEIC	S/262	rmusach@deic.uab.cat	937287757
Maria del Mar López	Seminaris i lab 1	dEIC	S/262	mmlopez@deic.uab.cat	937287757
Francesc Bosch	Lab 5 i 6	dEIC	S/260	fbosch@deic.uab.cat	937287757
Carles Caño	Lab 3 i 4	dEIC			937287757
Marta Pujol	Lab 7 i 8	dEIC	S/262	mpujol@deic.uab.cat	937287757

Coordinadors: Pere Guitart i Ramon Musach**Objectius**

Coneixements

- Conèixer i assimilar els conceptes fonamentals de l'algorísmica mitjançant l'aprenentatge i la comprensió de la sintaxi i la semàntica d'una notació algorísmica.
- Conèixer i assimilar els conceptes fonamentals de la programació de computadors: algorisme, estructura de la informació i programa.
- Conèixer i assimilar conceptes, mètodes i tècniques per tal que, a partir de l'enunciat d'un problema, es pugui especificar el comportament precís que haurà de tenir la solució, dissenyar-la i implementar-ne el programa corresponent.
- Saber diferenciar entre els tipus dades simples i compostos. I entre aquests, els tipus estàtics i dinàmics.
- Estudiar les tècniques bàsiques de la composició d'accions (procediments i funcions) i els mecanismes de comunicació.
- Estudiar els tipus de dades estructurats bàsics (taules, registres i cadenes de caràcters) i els tipus dinàmics (piles, cues, llistes i arbres binaris).
- Conèixer les tècniques bàsiques de disseny d'algorismes: esquemes de recorregut i cerca, descomposició modular i introducció a les tècniques de disseny recursiu.
- Introduir el problemes fonamentals de cerca i classificació. Estudiar i analitzar els algorismes bàsics de cerca (lineal i binària), els algorismes bàsics d'ordenació (selecció, inserció i intercanvi) i alguns d'avançats (ShellSort, QuickSort i HeapSort).
- Adquirir un coneixement no tan sols conceptual sinó també pràctic de com escriure un programa, compilar-lo, muntar-lo i executar-lo.
- Adquirir pràctica en l'aplicació de tots aquests conceptes dintre d'un entorn real de programació en el paradigma de la programació estructurada.

Habilitats

- Entendre i interpretar enunciats de problemes computacionals senzills.
- Treballar amb un llenguatge algorímic de propòsit general. Descriure l'algorisme per resoldre un problema computacional.
- Saber triar el millor algorisme per resoldre un problema entre diverses alternatives.
- Saber triar la millor forma de representació de la informació associada a un problema.
- Dissenyar algorismes d'acord amb les tècniques bàsiques de disseny d'algorismes.
- Desenvolupar programes en llenguatge C dintre d'un entorn real de desenvolupament.
- Realitzar petits projectes de programació en què s'hagi de demostrar la capacitat de dissenyar, codificar i analitzar la solució a un problema computacional de mida reduïda.
- Adquirir habilitats per presentar els raonaments i les conclusions amb claredat i precisió, i, de forma apropiada per a l'audiència a qui van dirigits, tant oralment com per escrit.
- Resumir, sintetitzar i interpretar textos tècnics de mida reduïda relacionats amb l'assignatura.

Competències

- Capacitat d'anàlisi i de síntesi.
- Comunicació oral i escrita.
- Treball en equip.
- Resolució de problemes.
- Raonament crític.
- Motivació i preocupació per la qualitat.
- Aprenentatge autònom.
- Capacitat d'abstracció.
- Capacitat d'aplicar els coneixements a la pràctica.

Capacitats prèvies

Aquesta assignatura no té cap prerequisit però és convenient que l'estudiant:

- Tingui un coneixement suficient d'aritmètica bàsica.
- Cursi o hagi cursat les assignatures Fonaments de Computadors i Àlgebra.

Continguts

(T: teoria, S: seminaris, PS: preparació de seminaris, L: laboratoris, PP: preparació pràctiques, E: estudi, AA: altres activitats)

Temes	Hores de dedicació de l'estudiant							Total
	T	S	PS	L	PP	E	AA	
0. Presentació de l'assignatura.	1							1
Presentació de l'assignatura, els materials, les activitats i la metodologia docent.								
1. Introducció a l'algorísmica.	3	6	12	6	6	3		36
Algorísmica. Objectes elementals. Expressions. Accions elementals. Composició d'accions. Sessió 1 Lab. L'entorn de treball. Exemples de programes bàsics en C. Sessió 2 Lab. Introducció al C: Constants, variables i operadors. Sessió 3 Lab. Introducció al C: Estructures de control.								

2. Metodologia dels esquemes seqüencials	T	S	PS	L	PP	E	AA	Total
	2	4	8	2	2	2		20
Esquemes seqüencials de recorregut i de cerca. Exemples d'utilització dels esquemes. Sessió 4 Lab. Esquemes seqüencials: recorregut i cerca.								

3. Algorismes i subalgorismes.	T	S	PS	L	PP	E	AA	Total
	2	4	8	2	2	2		20
Accions compostes: procediments i funcions. Àmbit d'un objecte. Pas de paràmetres. Sessió 6 Lab: Funcions i pas de paràmetres.								

4. Introducció als tipus de dades estructurats.	T	S	PS	L	PP	E	AA	Total
	3	6	12	4	4	3		32
Tipus de dades compostos: taules, registres i seqüències de caràcters. Sessió 7 Lab: Tipus de dades compostos I Sessió 8 Lab: Tipus de dades compostos II								

5. Cerca i classificació	T	S	PS	L	PP	E	AA	Total
	2	4	8	2	2	2		20
Algorismes de cerca. Algorismes directes de classificació. Introducció a l'anàlisi d'algorismes. Sessió 9 Lab: Entrada i Sortida de dades.								

6. El tipus simple punter. Memòria dinàmica	T	S	PS	L	PP	E	AA	Total
	2	4	8	4	4	2		24
El tipus simple punter. Memòria dinàmica. Declaració, construcció i destrucció d'objectes dinàmics. Sessió 11 Lab: Punters (tipus simples). Sessió 12 Lab: Punters (tipus compostos) i memòria dinàmica.								

7. Estructures dinàmiques: piles, cues i llistes	T	S	PS	L	PP	E	AA	Total
	4	8	16	4	4	4		40
Estructures de dades dinàmiques. Representacions. Piles. Cues. Aplicacions al càlcul d'expressions aritmètiques. Llistes, llistes circulars i llistes doblement enllaçades. Sessió 13 Lab: Piles i Cues. Sessió 14 Lab: Llistes.								

8. Fitxers	T	S	PS	L	PP	E	AA	Total
	2	4	8	2	2	2		20
Tipus de fitxers, suport, organització i accés. Fitxers seqüencials. Constructor i operacions associades. Els fitxers de text. Sessió 16 Lab: Fitxers (repàs i fitxers binaris)								

9. Algorismes recursius	T	S	PS	L	PP	E	AA	Total
	2	4	8	2	2	2		20
El concepte de recursivitat. Definicions recursives. Mecanisme de la recursivitat. Aplicacions de la recursivitat: QuickSort. Sessió 17 Lab: Recursivitat.								

10. Arbres.	T	S	PS	L	PP	E	AA	Total
	2	4	8			2		16
Definicions i propietats. Arbres binaris: representació i operacions bàsiques.								

11. Projecte de programació	T	S	PS	L	PP	E	AA	Total
				6	20			26

Sessió 18 Lab: Desenvolupament del projecte de programació
Sessió 19 Lab: Desenvolupament del projecte de programació
Sessió 20 Lab: Lliurament

12. Activitats d'avaluació	T	S	PS	L	PP	E	AA	Total
							28	28

Proves de validació 1, 2, 3 i 4.
Lliurament d'exercicis de pràctiques 1, 2 i 3.
Prova de pràctica 1, 2 i 3.

Total primer semestre: 128 h.

Total segon semestre: 146 h.

Activitats d'avaluació: 28 h.

Total: 302 hores.

Cicle d'aprenentatge

El cicle d'aprenentatge de l'assignatura es basa en una distribució setmanal d'activitats presencials i no presencials. Un model recomanable seria:

- Sessió teòrica. 1 hora presencial.
- Estudi i realització/revisió dels exercicis i activitats proposades: 3 hores
- Seminari: 2 hores presencials.
- Preparació sessió de laboratori: 1 hora.
- Laboratori: 2 hores presencials.
- Fer exercicis de programació: 1 hora.

Aquest model representa una dedicació tipus setmanal màxima de l'alumne de 10 hores, 5 de les quals són presencials. Aquesta dedicació màxima tindrà lloc les setmanes (de dimecres a dimarts) en les quals coincideixen les sessions de laboratori, de teoria i de seminari. Quan no hi ha sessions presencials, però, el temps necessari per a l'estudi i la realització d'exercicis acostuma a ser més gran. Finalment remarcar que en les darreres setmanes del curs el projecte de programació ocuparà la major part del temps. Com tot projecte de programació, els imprevistos que acostumen a donar-se poden puntualment generar una dedicació intensiva.

Metodologia docent

D'acord amb l'objectiu de centrar el procés d'aprenentatge en l'alumne es proposa una metodologia docent en la qual l'alumne (individualment o en grup) dirigeix el seu aprenentatge. Aquest procés es basarà en la metodologia següent.

L'alumne disposarà en el portfoli de l'assignatura situat en el Campus Virtual del material docent amb antel·lació suficient: contingut teòric, activitats obligatòries, activitats recomanades i la programació d'aquestes activitats.

1. **Exposició teòrica:** en la sessió de dimecres, el professor de teoria explicarà els objectius de cada tema, els continguts bàsics i les activitats (individuals o cooperatives) que cal realitzar per assolir els coneixements i les habilitats necessàries. També donarà les estratègies d'estudi i la bibliografia i el material necessari per assolir els objectius del tema. A partir d'aquesta informació l'alumne ha de planificar l'estudi de l'assignatura i la resolució dels exercicis i/o activitats proposades. Generalment, **es proposaran diversos exercicis d'avaluació que caldrà resoldre, moltes vegades en grup, i lliurar a l'inici del proper seminari.**

2. **Seminaris:** Les sessions de seminari són sessions de dues hores amb grups reduïts amb l'objectiu d'aprofundir en la comprensió de cada unitat docent. D'una banda, els estudiants hauran de lliurar i exposar els exercicis d'avaluació proposats, que es discutiran públicament. De l'altra, es proposaran i realitzaran noves activitats per tal de consolidar i/o ampliar la informació de la unitat docent. Bona part dels exercicis d'avaluació es faran en equips de treball. Cada equip tindrà a la seva disposició un *wiki* privat per tal de coordinar les tasques i incloure les solucions dels exercicis. Aquestes sessions, que haurien de servir per promoure, principalment, la capacitat d'anàlisi i de síntesi, la comunicació oral i escrita, el raonament crític, la resolució d'exercicis i el treball en equip, tenint un pes decisiu en l'avaluació de l'estudiant.
3. **Laboratoris:** Les sessions de laboratori duren dues hores i parteixen d'un guió que exposa algun aspecte relacionat en la programació en C, i que ha estat ja estudiat, o està essent treballat en paral·lel, en les sessions teòriques. Els guions contenen diversos exemples i una sèrie d'exercicis de programació que cal fer. En aquest punt es posarà especial atenció a la resolució de problemes i la motivació per a la qualitat.
4. **Projecte de programació:** El projecte de programació permetrà a l'alumne posar en pràctica tots els coneixements adquirits en l'assignatura. El projecte de programació es farà en el llenguatge de programació C i, al final, l'alumne haurà de demostrar tant coneixements sobre el llenguatge com sobre les diverses tècniques de programació que s'han treballat. En aquest punt es posarà especial atenció al treball en equip, la resolució de problemes i la motivació per a la qualitat.

Avaluació

Criteris d'avaluació:

L'avaluació de l'assignatura és continuada i té en compte tant els coneixements, com les competències i habilitats assolides. S'avaluaran especialment la constància i el progrés de l'alumne al llarg del curs. En aquest sentit es proposaran activitats d'avaluació obligatòries, però també n'hi haurà de voluntàries. En l'avaluació de cada activitat es tindran en compte no només els coneixements o competències que pretén assolir sinó també els indicadors de qualitat de l'activitat. Per exemple, en alguns exercicis de disseny d'algorismes es poden tenir en compte els criteris següents per establir la nota:

1. Correcció funcional de l'algorisme, és a dir, si la solució proposada resol el problema plantejat (50%)
2. Correcció de l'especificació de l'algorisme, és a dir, si la descripció (p.e. el llenguatge algorímic), estructura (capçalera, declaracions,...) i estil (comentaris, sagnat,...) són correctes (20%)
3. Correcció del disseny de l'algorisme, és a dir, la utilització adequada de recursos (estructures de dades, accions,...), la claredat, la generalitat, la correcció de l'algorisme,... (30%)

Activitats i instruments d'avaluació:

L'avaluació de l'assignatura es basa en la valoració de les que es proposaran en el curs:

- Resolució dels **exercicis d'avaluació** de cada tema que cal lliurar a l'inici del seminari.
- Resolució durant els seminaris dels **exercicis d'avaluació** proposats.
- Quatre **proves de validació**.
- Resolució, lliurament i defensa d'exercicis de codificació en C, de les sessions de laboratori.

- **Tres proves pràctiques de programació.**
- **Un projecte final de programació**

Indicadors i valoració:

Al llarg del curs i de manera continuada l'estudiant va acumulant punts en funció de les activitats que va realitzant. Com a molt pot obtenir 100 punts, 60 en la part de seminaris/teoria i proves de validació (49 dels quals s'obtenen mitjançant la realització dels exercicis d'avaluació de la part de teoria i 11 a partir de les proves de validació) i 40 en la part de pràctiques/laboratori. Per superar l'assignatura cal aconseguir un mínim de 50 punts sobre 100. D'aquests, **com a mínim 30** han de ser **punts de teoria** validats (sense tenir en compte els punts addicionals de les proves de validació) i **20 de les sessions pràctiques**. En les pràctiques, a més, **és obligatori presentar i aprovar el projecte final de programació**.

- Teoria: els punts de teoria s'obtenen amb la realització i/o exposició dels exercicis obligatoris i les activitats que es proposin. En general, per a cada tema de teoria es poden obtenir fins a 3 punts. **Per a que els punts d'un tema computin en la nota final, cal superar la prova de validació que inclou el tema**. Les activitats voluntàries permeten obtenir punts addicionals. El nombre màxim de punts és de 49 . Cas que es superi aquest màxim, només 49 punts es tindran en compte per al càlcul de la nota final.
- Proves de validació: les proves de validació tenen com a objectiu comprovar que l'estudiant, amb la feina que ha portat a terme, ha assolit el nivell de coneixements que es considera suficient. Podríem dir, per tant, que es tracta d'una prova de mínims. Amb l'objectiu de **promoure l'excel·lència** es puntuarà amb 2,5 punts addicionals les proves en què es mostrin coneixements amplis del tema (no de mínims). Amb l'objectiu de promoure la constància, els estudiants que superin les 4 proves de validació en primera convocatòria tindran 5 punts addicionals. El nombre màxim de punts en aquest apartat és d'11. Cas que es superi aquest màxim, només 11 punts es tindran en compte per al càlcul de la nota final.
- Pràctiques: dels 40 punts, 16 corresponen al projecte final de programació, 12 a les tres proves d'avaluació, 4 per a cadascuna, que es faran en horari de laboratori i 12 a les activitats obligatòries que es realitzen durant les sessions de pràctiques.

Qualificació final:

- *No presentat*: no ha participat en cap activitat d'avaluació posterior a la primera prova de validació.
- *Suspens*: menys de 30 punts de teoria o menys de 20 punts de pràctiques.
- *Aprobat*: de 50 a 69,9 punts.
- *Notable*: de 70 a 89,9 punts.
- *Excel·lent*: 90 o més punts.
- *Matrícula d'honor*: 90 punts o més i una de les tres millors notes del grup.

Segona convocatòria: Només per als alumnes que no hagin aprovat en primera convocatòria.

- Teoria: poden presentar-se a les proves de validació d'aquelles parts de l'assignatura que no hagin pogut validar. Només podran validar els punts obtinguts en l'avaluació continuada que tinguin pendents de validar.
- Pràctiques: es proposarà un nou projecte de programació per a aquells estudiants que no hagin aprovat el projecte de programació en primera convocatòria.

Calendari: Consulteu el calendari en el Portfoli de l'assignatura.

Dates previstes per a les proves de validació:

1a. prova de validació: dimarts 25/11/2008, a les 10.

2a. prova de validació: dimecres 04/02/09, a les 10.

3a. prova de validació : dijous 30/04/09, a les 10.

4a. prova de validació: divendres 26/06/09, a les 10.

Segona convocatòria: dilluns 06/07/09, a les 10.

Avaluació continuada	Examen final	2 ^a convocatòria
<input type="checkbox"/> NO	<input checked="" type="checkbox"/> No n'hi ha	<input type="checkbox"/> No n'hi ha
<input checked="" type="checkbox"/> SI En què consisteix? Resoldre i lliurar exercicis d'avaluació; resoldre exercicis addicionals durant el seminari; fer exercicis de codificació durant les sessions de laboratori; quatre proves de validació; tres proves d'avaluació de laboratori i un projecte final de programació.	<input type="checkbox"/> SI En què consisteix?	<input checked="" type="checkbox"/> Només per als alumnes que satisfacin els requisits següents: Alumnes que no hagin superat alguna de les proves de validació (i que tinguin punts per validar) o que no hagin aprovat el projecte de programació.
	<input type="checkbox"/> Obligatori per a tots	<input type="checkbox"/> Oberta a tots

Bibliografia bàsica

1. Aho, A.V. and Ullman, J.D. *Foundations of Computer Science*. Computer Science Press. 1992.
2. Antonakos, J.L. *Programación estructurada en C*. Ed. Prentice Hall, 1997.
3. Allen, M. *Estructuras de datos y algoritmos*. Addison-Wesley. 1995.
4. Botella López, Pere. *Fonaments de programació*. Ediuoc. Manuals 39. 2001.
5. Cormen, T.H.; Leiserson, C.E. and Rivest, R.L. *Introduction to Algorithms*. The MIT Press. 1992.
6. García, F.; Calderon, A.; Carretero, J.; Fernández, J.; Pérez, J.M. *Problemas resueltos de programación en lenguaje C*. Thompson, 2003.
7. García, J.J.; Montoya, F.J.; Fernández, J.L.; Majado, J.L. *Una introducción a la programación. Un enfoque algorítmico*. Thomson. 2005.
8. Brian W. Kernighan, Dennis Ritchie, Dennis M. Ritchie. *The C Programming Language* (2nd Edition), Prentice Hall.
9. Pujol, J. *Algorismes i programes*. Servei de publicacions UAB. Materials 27. 1998.
10. Wirth, N. *Algoritmos y Estructuras de datos*. Castillo. 1986.

Bibliografia complementària

1. Brassard, Gilles; Bratley, P. *Fundamentos de algoritmia*. Prentice Hall, 1997.
2. Castro, J.; Cucker, F. *et alt. Curs de programació*. McGraw-Hill. 1992.
3. Dijkstra, E.W. *A Method of programming*. Addison-Wesley, 1988.
4. Dromey, T.G. *How to solve it by computer*. Prentice Hall. 1982.
5. Knuth, D.E. El arte de programar ordenadores. Volumen I: *Algoritmos fundamentales*. 1986. Volumen III: *Clasificación y búsqueda*. 1987. Editorial Reverté.
6. Pratt, T.W.; Zelkowitz, M.V. *Lenguajes de programación. Diseño e implementación*. Prentice Hall. 1998.

Enllaços web

Aula Virtual de l'Autònoma Interactiva: <http://cv.uab.cat>

Web de l'EUI: <http://eui.uab.cat>

Wiki: <http://wiki.uab.cat>