

Metodologia de la Programació

Codi: 102764

Crèdits: 6

Titulació	Tipus	Curs	Semestre
2502441 Enginyeria Informàtica	FB	1	2

Professor/a de contacte

Nom: Ernest Valveny Llobet

Correu electrònic: ernest.valveny@uab.cat

Idiomes dels grups

Podeu accedir-hi des d'aquest [enllaç](#). Per consultar l'idioma us caldrà introduir el CODI de l'assignatura. Tingueu en compte que la informació és provisional fins a 30 de novembre de 2023.

Equip docent

Joan Sorribes Gomis

Prerequisits

L'assignatura no té cap prerequisit oficial. De totes formes, s'assumeix que l'estudiant ha cursat l'assignatura prèvia de Fonaments d'Informàtica i, per tant, està familiaritzat amb les estructures bàsiques de la programació.

Objectius

Aquesta assignatura forma part de la matèria Informàtica i s'ha de veure com la continuació lògica de la part de programació de l'assignatura Fonaments d'Informàtica. L'objectiu bàsic és aprofundir en les nocions bàsiques de la programació imperativa introduïdes a Fonaments d'Informàtica i introduir els principis de la programació orientada a objectes.

D'aquesta forma, els objectius formatius que es proposen per a l'assignatura són els següents:

- Entendre el cicle de vida del software: analitzar el problema (entendre el que se'ns demana), disseny (proposar una solució al problema), implementació (codificació en un llenguatge de programació de la solució escollida), prova (realització d'un test de manera sistemàtica per assegurar la correctesa de la solució implementada).
- Entendre el concepte d'algorisme com a eina de resolució de problemes amb l'ordinador i aprendre els conceptes fonamentals de l'algorísmica.
- Comprendre les principals estructures de la programació imperativa i utilitzar-les correctament per resoldre problemes algorísmics de certa complexitat.
- Conèixer les diferents estructures per representar informació dins dels algorismes, tant els tipus de dades estàtics (taules, registres i cadenes de caràcters) com els tipus de dades dinàmics (piles, cues,

l·listes), per poder utilitzar l'estructura de dades més adient per representar la informació associada un problema algorísmic.

- Entendre i aplicar correctament els principis bàsics de la programació orientada a objectes: concepte de classe i encapsulament de dades.
- Dotar l'alumne de la capacitat de disseny d'algorismes per a la resolució de problemes complexos, introduint de manera progressiva i sistemàtica una metodologia rigorosa i estructurada de programació, basada fonamentalment en la tècnica del disseny descendent d'algorismes.
- Programar en un llenguatge de programació real i conèixer les diferents fases de desenvolupament d'un programa: escriptura, compilació i muntatge i execució i prova.
- Desenvolupar els programes seguint unes normes d'estil tendents a aconseguir programes de qualitat. Dins d'aquestes normes d'estil s'engloben aquelles que faciliten la comprensió del codi, com poden ser l'ús de comentaris, la indentació del codi, la utilització de noms adequats per variables i funcions, etc.

Competències

- Adquirir hàbits de pensament.
- Adquirir hàbits de treball personal.
- Capacitat per comprendre i dominar els conceptes bàsics de matemàtica discreta, lògica, algorítmica i complexitat computacional, i la seva aplicació per a la resolució de problemes propis de l'enginyeria.
- Coneixements bàsics sobre l'ús i programació de computadors, sistemes operatius, bases de dades i programes informàtics amb aplicació en enginyeria.
- Conèixer les matèries bàsiques i les tecnologies que capacitin per a l'aprenentatge i el desenvolupament de nous mètodes i tecnologies, així com d'aquelles que els dotin d'una gran versatilitat per a adaptar-se a noves situacions.

Resultats d'aprenentatge

1. Comprendre i dominar l'algorítmica, la complexitat computacional i la seva aplicació a la resolució de problemes.
2. Conèixer els principis bàsics de l'estructura i la programació dels computadors.
3. Conèixer i ser capaç d'utilitzar sistemes operatius, bases de dades i programes d'ús comú en l'enginyeria.
4. Demostrar capacitat per a l'organització de la informació en fitxers i bases de dades.
5. Demostrar capacitat per a la representació de la informació i la programació de computadors.
6. Demostrar la capacitat per a dissenyar algorismes i analitzar la seva complexitat computacional.
7. Desenvolupar un mode de pensament i raonament crítics.
8. Gestionar el temps i els recursos disponibles. Treballar de manera organitzada.
9. Prendre decisions pròpies.
10. Reconèixer i identificar els mètodes, sistemes i tecnologies propis de l'enginyeria informàtica.

Continguts

Tema 0: Repàs d'estructures de programació bàsiques

- Repàs de les estructures de programació bàsiques: estructures condicionals i iteratives, funcions i procediments, taules, matrius, registres

Tema 1: Introducció a la programació orientada a objectes

- Introducció al concepte de classe. Mètodes i atributs. Part privada i pública.
- Constructors i destructors.

- Encapsulament de dades.
- Composició de classes.
- Persistència i serialització d'objectes. Fixters.

Tema 2: Estructures de dades dinàmiques

- El concepte d'apuntador. Operacions amb apuntadors.
- Objectes dinàmics.
- Arrays dinàmics
- Representació i implementació d'estructures de dades dinàmiques: llistes, piles i cues.
- Utilització d'estructures de dades dinàmiques. Llibreria STL

Tema 3: Anàlisi de costos i de complexitat

Metodologia

La metodologia docent de l'assignatura parteix del principi que diu que "**programar és l'única forma d'aprendre a programar**" i, per tant, estarà centrada principalment en el treball pràctic de l'estudiant. Les sessions presencials de classe s'organitzaran per introduir els continguts teòrics de l'assignatura, des d'una perspectiva molt pràctica a partir d'exemples i d'exercicis i problemes de programació que s'hauran de resoldre a classe directament amb l'ordinador. Per una altra banda, es realitzarà un projecte de programació que s'haurà d'anar desenvolupant de forma principalment autònoma durant tot els curs (amb seguiment i control per part del professor en sessions puntuals) i que suposarà integrar de forma pràctica gairebé tots els conceptes i eines de programació introduïts a les sessions presencials en la resolució d'un problema real complex. A més a més, es proposarà un conjunt d'exercicis que s'hauran de resoldre de forma individual al llarg del curs (alguns dels quals s'avaluaran i discutiran a les sessions presencials) que han de servir per comprendre, integrar i aplicar els conceptes desenvolupats a les sessions presencials. A totes les activitats del curs (sessions presencials, problemes i pràctiques) es farà servir el llenguatge de programació C++.

A nivell presencial, l'assignatura no distingirà entre sessions de teoria, problemes i pràctiques. Les sessions presencials s'organitzaran en quatre hores setmanals en grups al voltant de 40 estudiants. A les sessions presencials s'aniran treballant els conceptes que es detallen al temari de l'assignatura. En alguns casos, es podran posar a disposició de l'estudiant vídeos explicatius que l'estudiant haurà de visionar abans de la sessió de classe. Les sessions de classe tindran un enfoc força pràctic amb exemples i exercicis que es plantejaran als alumnes per facilitar la comprensió i aprenentatge dels conceptes explicats. Aquests exercicis es realitzaran i discutiran durant la sessió i serviran per anar introduint els continguts de l'assignatura i veure'n la seva aplicació pràctica. S'encoratja que l'alumne porti el seu propi portàtil a classe si en disposa d'un per poder realitzar els exercicis que es proposin.

L'estudiant haurà de completar les classes presencials amb el treball personal autònom en la realització dels exercicis que es vagin proposant i que han de servir per acabar d'entendre els continguts de l'assignatura. Cal tenir present que el temari de l'assignatura té una continuïtat lògica al llarg del curs, de manera que per poder seguir correctament una classe cal haver assimilat el que s'ha explicat a les sessions anteriors. Alguns d'aquests exercicis s'hauran de lliurar de forma individual com a part de l'avaluació de l'assignatura.

A més a més, els estudiants hauran de fer en grups de 2 un projecte de programació que es desenvoluparà de forma autònoma durant tot el curs fora de les sessions presencial. El projecte de programació permetrà abordar un problema de programació d'una certa complexitat que integri la majoria dels conceptes explicats durant el curs. Durant el curs, es dedicaran algunes sessions presencials al control, seguiment i avaluació del treball fet per l'alumne en el projecte de programació.

La gestió de la docència l'assignatura es farà a través de la plataforma Caronte (<http://caronte.uab.cat/>), que servirà per poder veure els materials, gestionar els grups de pràctiques, fer els lliuraments corresponents, veure les notes, comunicar-se amb els professors, etc.

Competències Transversals

- T01.01 Desenvolupar un pensament i raonament crític. Aquesta competència es desenvoluparà durant les sessions de teoria i problemes, a partir de l'exposició i discussió d'exemples i casos pràctics. Es tindrà en compte a l'avaluació de l'exposició dels estudiants a les sessions de problemes.
- T02.03 Gestionar el temps i els recursos disponibles. Treballar de forma organitzada. Aquesta competència es desenvoluparà principalment a partir del projecte de programació. L'estudiant té com a objectiu desenvolupar el projecte de forma autònoma i ha de ser capaç d'organitzar-se el temps i els recursos per aconseguir aquest objectiu. S'avaluarà a les sessions de seguiment del projecte.
- T02.05 Prendre decisions pròpies. Aquesta competència es desenvoluparà durant el projecte de programació en què l'estudiant haurà d'escollir i prendre les millors opcions per aconseguir completar el projecte. S'avaluarà a les sessions de seguiment del projecte.

Nota: es reservaran 15 minuts d'una classe, dins del calendari establert pel centre/titulació, per a la complementació per part de l'alumnat de les enquestes d'avaluació de l'actuació del professorat i d'avaluació de l'assignatura/mòdul.

Activitats formatives

Títol	Hores	ECTS	Resultats d'aprenentatge
Tipus: Dirigides			
Sessions presencials	50	2	1, 2, 4, 5, 6, 7, 10
Tipus: Supervisades			
Seguiment de la resolució del projecte de programació	1	0,04	6, 7, 8, 9
Tipus: Autònomes			
Estudi individual	11	0,44	2, 10
Implementació del projecte de programació	48	1,92	3, 4, 5, 6, 8, 9
Resolució de problemes de manera individual i grupal	36	1,44	4, 5, 6, 8

Avaluació

L'avaluació de l'assignatura tindrà en compte tres tipus d'activitats d'avaluació: lliurament de problemes, avaluació individual i projecte de programació. La nota final de l'assignatura s'obté combinant l'avaluació d'aquestes 3 activitats de la manera següent:

$$\text{Nota Final} = (0.3 * \text{Avaluació continuada}) + (0.3 * \text{Projecte}) + (0.4 * \text{Avaluació Individual})$$

- Avaluació continuada: en aquest apartat s'inclou el lliurament dels exercicis que es proposin al llarg del curs i altres activitats que es realitzin a les sessions presencials o es proposin per lliurar a la plataforma Caronte.
 - S'haurà d'aconseguir una nota mínima de 4 en aquesta activitat per poder aprovar l'assignatura.
 - Els exercicis que no es lliurin en el termini fixat o que tinguin una avaluació de suspès es podran recuperar i tornar a lliurar en qualsevol moment del curs abans de la data de l'examen final de l'assignatura, amb una reducció sobre la nota del 20%.
- Avaluació individual: en aquest apartat s'inclou el resultat de les proves individuals que es faran al llarg del curs. Hi haurà dues proves parcials que es faran durant el període lectiu del curs en horaris de classe i una prova final durant el període oficial d'exàmens. Aquesta prova final serà de recuperació i

només l'hauran de fer els estudiants que no hagin superat algun dels dos parcials. Si s'ha superat un dels dos parcials, però l'altre no, en aquesta prova només s'ha de recuperar la part de l'assignatura corresponent al parcial que no s'hagi superat.

- S'haurà d'aconseguir una nota mínima de 4,5 en cadascun dels dos parcials per poder aprovar l'assignatura.
- La nota final serà la mitja dels dos parcials:

$$\text{Avaluació Individual} = (0.5 * \text{Parcial1}) + (0.5 * \text{Parcial 2})$$

- S'haurà d'aconseguir una nota mínima de 5 en la nota de l'avaluació individual per poder aprovar l'assignatura.
- Projecte: inclou tot el treball del projecte de programació. Inclou l'avaluació dels dos lliuraments del projecte (un lliurament parcial a meitat de curs i el lliurament final) i l'avaluació del seguiment del projecte que es farà al llarg del curs.
- S'haurà d'aconseguir una nota mínima de 5 en el lliurament final del projecte per poder aprovar el projecte.
- La nota de l'entrega final del projecte es podrà recuperar si la nota del projecte és ≥ 3 . A la recuperació del projecte la nota màxima que es podrà obtenir és un 7.

Avaluació única

L'estudiantat que es vulgui acollir a l'avaluació única haurà de lliurar en la data indicada per l'avaluació única els exercicis avaluable de curs que s'indiquin al principi de curs i el projecte de programació i haurà de realitzar una prova d'avaluació individual que inclourà els continguts de tot el curs. Aquesta prova d'avaluació individual constarà de dues parts que es correspondran als dos parcials que es faran al llarg del curs i s'haurà d'aconseguir un mínim de 4,5 a cadascuna de les dues parts per poder aprovar l'assignatura.

La nota final en el cas d'avaluació única es calcularà d'aquesta forma:

$$\text{Nota Final} = (0.2 * \text{Lliurament problemes}) + (0.3 * \text{Projecte}) + (0.5 * \text{Avaluació Individual})$$

Les notes mínimes de cada part per poder aprovar l'assignatura són les mateixes que per l'avaluació continuada.

S'aplicarà el mateix sistema de recuperació que per l'avaluació continuada

No avaluable: Un alumne es considerarà no avaluable (NA) si no fa com a mínim el 50% dels lliuraments d'exercicis i no fa cap de les proves d'avaluació següents: parcial 1, parcial 2, prova final de recuperació, lliurament final de la pràctica.

Suspesos: Si el càlcul de la nota final és igual o superior a 5 però no s'arriba al mínim exigint en alguna de les activitats d'avaluació, la nota final serà suspès i es posarà un 4.5 a la nota de l'expedient de l'alumne.

Matrícules d'honor: Atorgar una qualificació de matrícula d'honor és decisió del professorat responsable de l'assignatura. La normativa de la UAB indica que les MH només es podran concedir a estudiants que hagin obtingut una qualificació final igual o superior a 9.00. Es pot atorgar fins a un 5% de MH del total d'estudiants matriculats.

Convalidacions: Pels alumnes repetidors es podrà convalidar el projecte de l'any anterior si es compleixen aquestes condicions:

- La nota final del projecte del curs anterior és més gran o igual a 7
- La nota de l'avaluació individual del curs anterior és més gran o igual a 3

Si es convalida, la nota del projecte serà un 5, independentment de la nota obtinguda l'any anterior

Revisions de notes: Per a cada activitat d'avaluació, s'indicarà un lloc, data i hora de revisió en la que l'estudiant podrà revisar l'activitat amb el professor. En aquest context, es podran fer reclamacions sobre la nota de l'activitat, que seran avaluades pel professorat responsable de l'assignatura. Si l'estudiant no es presenta a aquesta revisió, no es revisarà posteriorment aquesta activitat.

Nota important: còpies i plagis

Sense perjudici d'altres mesures disciplinàries que s'estimin oportunes, i d'acord amb la normativa acadèmica vigent, les irregularitats comeses per un estudiant que puguin conduir a una variació de la qualificació es qualificaran amb un zero (0). Les activitats d'avaluació qualificades d'aquesta forma i per aquest procediment no seran recuperables. Si és necessari superar qualsevol d'aquestes activitats d'avaluació per aprovar l'assignatura, aquesta assignatura quedarà suspesa directament, sense oportunitat de recuperar-la en el mateix curs. Aquestes irregularitats inclouen, entre d'altres:

- la còpia total o parcial d'una pràctica, informe, o qualsevol altra activitat d'avaluació;
- deixar copiar;
- presentar un treball de grup no fet íntegrament pels membres del grup;
- presentar com a propis materials elaborats per un tercer, encara que siguin traduccions o adaptacions, i en general treballs amb elements no originals i exclusius de l'estudiant;
- tenir dispositius de comunicació (com telèfons mòbils, smart watches, etc.) accessibles durant les proves d'avaluació teórico-pràctiques individuals (exàmens);
- parlar amb companys durant les proves d'avaluació teórico-pràctiques individuals (exàmens);
- copiar o intentar copiar d'altres alumnes durant les proves d'avaluació teórico-pràctiques (exàmens);
- usar o intentar usar escrits relacionats amb la matèria durant la realització de les proves d'avaluació teórico-pràctiques (exàmens), quan aquests no hagin estat explícitament permesos.

En aquests casos, la nota numèrica de l'expedient serà el valor menor entre 3.0 i la mitjana ponderada de les notes (i per tant no serà possible l'aprobat per compensació).

En l'avaluació dels lliuraments de problemes i pràctiques s'utilitzaran eines de detecció de còpia del codi del programa.

Nota sobre la planificació de les activitats d'avaluació: Les dates d'avaluació continuada i lliurament de treballs es publicaran al principi de curs i poden estar subjectes a canvis de programació per motius d'adaptació a possibles incidències. Sempre s'informarà a Caronte sobre aquests canvis ja que s'entén que aquesta és la plataforma habitual d'intercanvi d'informació entre professors i estudiants.

Activitats d'avaluació continuada

Títol	Pes	Hores	ECTS	Resultats d'aprenentatge
Avaluació individual	40%	4	0,16	2, 10
Lliurament de problemes	30%	0	0	1, 4, 5, 6, 7, 8
Projecte de programació	30%	0	0	1, 3, 4, 5, 6, 8, 9

Bibliografia

- <http://www.cplusplus.com/> : The C++ Resources Network
- https://es.wikibooks.org/wiki/Programaci%C3%B3n_en_C%2B%2B: Programación en C++ - Wikilibros
- <https://www.sololearn.com/>: SoloLearn
- L. Joyanes, I. Zahonero: Programación en C: metodología, estructura de datos y objetos, Mc Graw-Hill, 2001.
- B. Eckel. Thinking in C++, Volume 1: Introduction to Standard C++, Prentice-Hall, 1999.
- B. Eckel. Thinking in C++, Volume 2: Standard Libraries and Advanced Topics, Prentice-Hall, 1999.
- F. Xhafa, P. Vázquez, J. Marco, X. Molinero, A. Martín: Programación en C++ para ingenieros. Thomson, 2006.

- E. Valveny, R. Benavente, A. Lapedriza, M. Ferrer, J. García: Programació en Llenguatge C. Amb 56 problemes resolts i comentats. Servei publicacions UAB, 2009.
- L. Joyanes, A. Castillo, L. Sánchez, I. Zahonero: Programación en C: libro de problemas, Mc Graw-Hill, 2002.
- B.W. Kernighan, D.M. Ritchie: El lenguaje de programación C. 2ª Edición, Prentice Hall, 1986.
- B.W. Kernighan, R. Pike: La Práctica de la Programación. Pearson Educación, 2000.
- M. Jesús Marco Galindo et al. Fonaments de Programació. Publicacions UOC, 2008.
- L. Joyanes Aguilar : Fundamentos de Programación: Algoritmos, Estructuras de Datos y Objetos. 3ª Edición, Mc. Graw-Hill, 2003.
- J. Pujol: Algorismes i Programes. Servei de Publicacions de la UAB, 1996.

Programari

Qualsevol entorn de desenvolupament en C++.