

Programació 2

Codi: 107890

Crèdits: 6

2025/2026

Titulació	Tipus	Curs
Enginyeria Informàtica	FB	1

Professor/a de contacte

Nom: Jorge Bernal del Nozal

Correu electrònic: jorge.bernal@uab.cat

Equip docent

Francisco Javier Massó López

Mireia Bellot Garcia

Idiomes dels grups

Podeu consultar aquesta informació al [final](#) del document.

Prerequisits

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

Objectius

Aquesta assignatura forma part de la matèria 3 (Informàtica) del Grau d'Enginyeria Informàtica i s'ha de veure com la continuació lògica de Programació I. L'objectiu bàsic és aprofundir en les nocions bàsiques de la programació imperativa introduïdes a Programació I 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 desenvolupament 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, llistes), 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.
- Ser capaç de fer un exercici d'abstracció per poder traslladar els conceptes teòrics de la assignatura, especialment els relatius a la programació orientada a objectes, a diferents llenguatges de programació, com poden ser C++ o Python

Resultats d'aprenentatge

1. CM04 (Competència) Validar les solucions proposades a problemes de l'àmbit de la informàtica en funció dels seus requisits i atenent les condicions establertes.
2. KM06 (Coneixement) Identificar els conceptes bàsics de la programació estructurada, modular i orientada a objectes, així com les eines i els entorns de desenvolupament per a la creació de programari de qualitat.
3. KM07 (Coneixement) Identificar els tipus de dades simples i les estructures de dades complexes dels llenguatges de programació d'alt nivell, així com el seu emmagatzematge en memòria.
4. SM09 (Habilitat) Analitzar les necessitats de programació d'un sistema informàtic des del punt de vista de la necessitat del client.
5. SM10 (Habilitat) Fer servir eines per a l'anàlisi, el disseny, la codificació i la depuració de programes.
6. SM11 (Habilitat) Desenvolupar programari de complexitat petita o mitjana per resoldre problemes utilitzant programació estructurada i orientada a objectes.
7. SM12 (Habilitat) Demostrar habilitats per treballar cooperativament en el desenvolupament d'aplicacions informàtiques.

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. Fitxers.

Tema 2: Estructures de dades dinàmiques

- 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
- Anàlisi de costos i complexitat

Activitats formatives i Metodologia

Títol	Hores	ECTS	Resultats d'aprenentatge
Tipus: Dirigides			
Estudi individual per preparar proves d'avaluació	11	0,44	KM06, KM07, SM09, SM11
Sessions de teoria i exercicis	50	2	CM04, KM06, KM07, SM09, SM10, SM11
Tipus: Supervisades			
Seguiment de la resolució del projecte de programació	1	0,04	CM04, SM09, SM10, SM11
Tipus: Autònomes			
Realització del projecte de programació	48	1,92	CM04, KM06, KM07, SM09, SM10, SM11, SM12
Resolució de problemes	36	1,44	CM04, KM06, KM07, SM10, SM11

La metodologia docent de l'assignatura parteix del principi que "programar és l'única manera d'aprendre a programar" i, per tant, se centrarà principalment en el treball pràctic de l'estudiant. Les sessions presencials s'organitzaran per introduir els continguts teòrics de l'assignatura des d'una perspectiva molt pràctica, a partir d'exemples, exercicis i problemes de programació que s'hauran de resoldre a classe directament amb l'ordinador. Un dels objectius de l'assignatura és separar els conceptes teòrics (com ara la definició d'una classe o el tractament de fitxers) de la seva implementació en un llenguatge de programació específic, tot i que el llenguatge vehicular de l'assignatura serà el C++.

A més de l'explicació dels conceptes teòrics i la resolució de petits problemes a mode d'exemple durant les classes, també es durà a terme un projecte de programació que s'haurà d'anar desenvolupant de manera principalment autònoma al llarg del curs (amb seguiment i control puntuals per part del professorat). Aquest projecte permetrà integrar de manera pràctica gairebé tots els conceptes i eines de programació introduïts a les sessions presencials en la resolució d'un problema real i complex.

Finalment, també es proposarà un conjunt d'exercicis i activitats complementàries (com ara qüestionaris) que s'hauran de resoldre de manera individual durant el curs. Alguns d'aquests exercicis s'avaluaran i es discutiran a les sessions presencials, i han de servir per comprendre, integrar i aplicar els conceptes desenvolupats a classe. A totes les activitats del curs (sessions presencials, problemes i pràctiques) s'utilitzarà principalment el llenguatge de programació C++, tot i que també s'explicarà de manera complementària com codificar els conceptes més importants utilitzant Python.

Pel que fa a la presencialitat, l'assignatura no distingirà entre sessions de teoria, problemes i pràctiques. Les sessions presencials s'organitzaran en quatre hores setmanals en grups d'uns 40 estudiants. Durant aquestes sessions es treballaran els conceptes detallats al temari de l'assignatura. En alguns casos, es podran posar a disposició de l'estudiant vídeos explicatius que caldrà visualitzar abans de la sessió. Les classes tindran un enfocament molt pràctic, amb exemples i exercicis que es plantejaran per facilitar la comprensió i l'aprenentatge dels conceptes explicats. Aquests exercicis es realitzaran i es discutiran durant la sessió, i serviran per introduir els continguts de l'assignatura i veure'n l'aplicació pràctica. Es recomana que l'estudiant porti el seu propi ordinador portàtil a classe, si en disposa, per poder fer els exercicis proposats.

L'estudiant haurà de complementar les classes presencials amb treball personal autònom, resolent els exercicis proposats, que han de servir per acabar d'assimilar els continguts de l'assignatura. Cal tenir present que el temari té una continuïtat lògica al llarg del curs, de manera que per 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 manera individual com a part de l'avaluació.

A més, els estudiants hauran de fer, en grups de dos, un projecte de programació que es desenvoluparà de manera autònoma durant tot el curs, fora de les sessions presencials. Aquest projecte permetrà abordar un problema de programació amb una certa complexitat, integrant la majoria dels conceptes explicats durant el curs. Al llarg del curs, es dedicaran algunes sessions presencials al control, seguiment i avaluació del treball realitzat.

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

Nota: es reservaran 15 minuts d'una classe, dins del calendari establert pel centre/titulació, perquè els alumnes compleixin les enquestes d'avaluació de l'actuació del professorat i d'avaluació de l'assignatura.

Avaluació

Activitats d'avaluació continuada

Títol	Pes	Hores	ECTS	Resultats d'aprenentatge
Activitats avaluables a realitzar a classe	10%	0	0	CM04, KM06, KM07, SM09, SM10, SM11
Primer examen parcial	30%	2	0,08	CM04, KM06, KM07, SM09, SM11
Projecte de programació	15%	0	0	CM04, KM06, KM07, SM09, SM10, SM11, SM12
Qüestionaris d'autoavaluació	5%	0	0	KM06, KM07
Resolució de problemes	10%	0	0	CM04, KM06, KM07, SM09, SM10, SM11, SM12
Segon examen parcial	30%	2	0,08	CM04, KM06, KM07, SM09, SM11

L'avaluació de l'assignatura es basarà en tres tipus d'activitats:

- Activitats d'avaluació continuada
- Avaluació individual mitjançant exàmens parcials
- Projecte de programació

La nota final de l'assignatura es calcularà combinant les tres activitats de la manera següent:

Nota Final = (0,25 × Avaluació continuada) + (0,15 × Projecte) + (0,6 × Avaluació individual)

1. Avaluació continuada (individual)

Aquest apartat inclou:

- Lliurament dels exercicis proposats a la plataforma Caronte (10%)
- Resolució i lliurament de petits problemes durant les sessions de classe (10%)
- Qüestionaris d'autoavaluació a Caronte, realitzats i lliurats durant les sessions de classe (5%)

No hi ha una nota mínima exigida en aquest apartat, però no es podran recuperar les activitats amb el termini de lliurament exhaurit. Es recomana participar activament en aquestes activitats, ja que són clau per assimilar els coneixements en temps real i proporcionen informació valuosa al professorat per adaptar les activitats, aprofundint en conceptes que no s'hagin assimilats correctament.

2. Avaluació individual (individual)

Inclou els resultats de les proves individuals realitzades durant el curs:

- Dues proves parcials durant el període lectiu, en horari de classe
- Una prova final durant el període oficial d'exàmens, només per a estudiants que no hagin superat algun dels dos parcials

Si només s'ha suspès un dels parcials, la prova final servirà per recuperar exclusivament la part corresponent.

Per aprovar l'assignatura, caldrà:

- Obtenir una nota mínima de 5 en cadascun dels dos parcials
- Obtenir una nota mínima de 5 en el càlcul global de l'avaluació individual

Càlcul de la nota d'avaluació individual:

Avaluació individual = $(0,5 \times \text{Parcial 1}) + (0,5 \times \text{Parcial 2})$

3. Projecte de programació (en grup i individual)

Inclou:

- Avaluació dels dos lliuraments del projecte (un parcial a meitat de curs i un final)
- Seguiment del projecte durant el curs
- Possibilitat de realitzar una prova de validació individual per garantir la participació de tots els membres del grup

Per aprovar el projecte caldrà:

- Obtenir una nota mínima de 5 en el lliurament final
- Superar la prova d'avaluació individual

Si la nota del lliurament final és igual o superior a 3, es podrà recuperar. En aquest cas, la nota màxima que es podrà obtenir serà un 7.

Avaluació única

L'estudiantat que opti per l'avaluació única haurà de:

- Lliurar, en la data indicada, els exercicis avaluables especificats a l'inici del curs
- Lliurar el projecte de programació
- Realitzar una prova d'avaluació individual que inclourà tot el contingut del curs

Aquesta prova constarà de dues parts, corresponents als dos parcials. Caldrà obtenir una nota mínima de 4,5 en cadascuna per poder aprovar.

Càlcul de la nota final en l'avaluació única:

Nota Final = $(0,2 \times \text{Lliurament de problemes a Caronte}) + (0,2 \times \text{Projecte}) + (0,6 \times \text{Avaluació individual})$

Les notes mínimes requerides per aprovar són les mateixes que en l'avaluació continuada. També s'aplicarà el mateix sistema de recuperació.

Casos especials de notes finals

No evaluable:

Un alumne es considerarà *no evaluable* (NA) si no participa en cap de les activitats avaluables de l'assignatura.

Suspès:

En cas de no superar l'assignatura perquè alguna de les activitats d'avaluació no assoleix la nota mínima

requerida, la qualificació numèrica que constarà a l'expedient serà el valor menor entre 4,5 i la mitjana ponderada de les notes.

Matrícules d'honor:

L'atorgament de la qualificació de *matrícula d'honor* és una decisió del professorat responsable de l'assignatura. La normativa de la UAB estableix que només es poden concedir MH a estudiants que hagin obtingut una qualificació final igual o superior a 9,00. Es pot atorgar fins a un 5% de MH sobre el total d'estudiants matriculats. El criteri utilitzat per concedir-les serà la nota final de l'assignatura.

Revisions de notes

Per a cada activitat d'avaluació, s'indicarà el lloc, la data i l'hora de revisió, durant la qual l'estudiant podrà revisar l'activitat amb el professorat. En aquest context, es podran presentar reclamacions sobre la qualificació obtinguda, que seran valorades pel professorat responsable de l'assignatura. Si l'estudiant no assisteix a la revisió en el moment establert, no es podrà revisar posteriorment l'activitat corresponent.

Alumnat repetidor

Aquesta assignatura no contempla un tractament diferenciat per a l'alumnat repetidor. No es convalidarà cap projecte d'anys anteriors, ja que es tracta d'una assignatura nova, amb un equip docent i una metodologia renovats.

Nota important: còpies i plagis

Sense perjudici d'altres mesures disciplinàries que es considerin oportunes, i d'acord amb la normativa acadèmica vigent, qualsevol irregularitat comesa per un estudiant que pugui afectar la qualificació d'una activitat serà sancionada amb una nota de zero (0). Aquestes activitats no seran recuperables. Si l'activitat en qüestió és necessària per aprovar l'assignatura, aquesta quedarà suspesa automàticament, sense possibilitat de recuperació durant el mateix curs.

Les irregularitats inclouen, entre d'altres:

- Còpia total o parcial d'una pràctica, informe o qualsevol altra activitat d'avaluació
- Permetre que altres estudiants copiïn
- Presentar un treball de grup no elaborat íntegrament pels membres del grup
- Presentar com a propis materials elaborats per tercers, incloent-hi traduccions o adaptacions
- Utilitzar dispositius de comunicació (mòbils, rellotges intel·ligents, etc.) durant les proves d'avaluació teòrico-pràctiques
- Comunicar-se amb altres estudiants durant les proves
- Copiar o intentar copiar d'altres alumnes
- Utilitzar o intentar utilitzar materials relacionats amb la matèria durant les proves, si no han estat explícitament autoritzats

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 aprovar per compensació.

Per a l'avaluació dels lliuraments de problemes i pràctiques, s'utilitzaran eines de detecció de còpia de codi.

Revisions de qualificacions

Les dates de les activitats d'avaluació continuada i dels lliuraments de treballs es publicaran a l'inici del curs. Aquestes dates poden estar subjectes a modificacions per motius d'adaptació a possibles incidències. Tots els canvis es comunicaran a través de la plataforma Caronte, que s'entén com el canal habitual d'intercanvi d'informació entre professorat i alumnat.

Ús de la Intel·ligència Artificial (IA)

En aquesta assignatura es permet l'ús de tecnologies d'IA exclusivament com a suport, per exemple per a la cerca d'informació, la interpretació de programes, la resolució de dubtes, etc. No es permet l'ús de la IA per a la realització de les tasques de programació que s'han de lliurar com a part de l'avaluació.

Qualsevol treball que inclogui fragments generats amb IA serà considerat una falta d'honestat acadèmica i es tractarà com una irregularitat segons la normativa vigent, podent suposar el suspens a l'assignatura.

Consideracions addicionals

Les situacions següents s'han de gestionar a través de Gestió Acadèmica de l'Escola d'Enginyeria, que informará l'estudiant sobre els protocols i normatives aplicables:

- Sol·licitud de traducció del material d'avaluació a un idioma diferent del català
- Petició de reprogramació de proves d'avaluació
- Sol·licitud per acollir-se a l'avaluació única

El professorat no té potestat per prendre decisions sobre aquests aspectes i es limitarà a aplicar les resolucions dels òrgans competents de l'Escola.

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.
- 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.
- L. Joyanes Aguilar : Fundamentos de Programación: Algoritmos, Estructuras de Datos y Objetos. 3ª Edición, Mc. Graw-Hill, 2003
- J. Guttag. Introduction to Computation and Programming Using Python: With Application to Understanding Data. Second Edition. MIT Press. ISBN-10: 9780262529624.
- S. Chazallet Python 3. Los fundamentos del lenguaje. Eni, ISBN-10: 2409006140.
- Steven F. Lott. Mastering object-oriented Python. Packt publishing, 2014.

Programari

Qualsevol entorn de desenvolupament en C++ y Python (Visual Studio Code)

Grups i idiomes de l'assignatura

La informació proporcionada és provisional fins al 30 de novembre de 2025. A partir d'aquesta data, podreu consultar l'idioma de cada grup a través d'aquest [enllaç](#). Per accedir a la informació, caldrà introduir el CODI de l'assignatura

Nom	Grup	Idioma	Semestre	Torn
(PAUL) Pràctiques d'aula	411	Català/Espanyol	segon quadrimestre	matí-mixt

(PAUL) Pràctiques d'aula	412	Català/Espanyol	segon quadrimestre	matí-mixt
(PAUL) Pràctiques d'aula	431	Català/Espanyol	segon quadrimestre	matí-mixt
(PAUL) Pràctiques d'aula	432	Català/Espanyol	segon quadrimestre	matí-mixt
(PAUL) Pràctiques d'aula	451	Català/Espanyol	segon quadrimestre	tarda
(PAUL) Pràctiques d'aula	452	Català/Espanyol	segon quadrimestre	matí-mixt
(PAUL) Pràctiques d'aula	471	Català/Espanyol	segon quadrimestre	matí-mixt
(PAUL) Pràctiques d'aula	472	Català/Espanyol	segon quadrimestre	tarda