

Fonaments de Programació II

Codi: 106554
Crèdits: 6

Titulació	Tipus	Curs	Semestre
2504392 Intel·ligència Artificial / Artificial Intelligence	FB	1	2

Professor/a de contacte

Nom: Anna Bàrbara Sikora
Correu electrònic: anna.sikora@uab.cat

Utilització d'idiomes a l'assignatura

Llengua vehicular majoritària: anglès (eng)
Grup íntegre en anglès: Sí
Grup íntegre en català: No
Grup íntegre en espanyol: No

Prerequisits

El curs no té requisits previs oficials. Tot i això, es basa en Fonaments de Programació I. Per tant, es recomana que els/les estudiants tinguin un bon coneixement dels conceptes bàsics de programació i Python.

Objectius

Aquest curs és la continuació de Fonaments de Programació I, i els objectius generals són:

- Comprendre diferents estructures de dades, com piles, cues i llistes.
- Comprendre diferents estructures de dades més complexes, com taules hash, arbres i gràfics.
- Comprendre la memòria dinàmica i els punters.
- Comprendre diferents algorismes per ordenar i cercar, i les limitacions d'aquests algorismes.
- Ser capaç d'analitzar un problema complex, dissenyar una solució eficient, implementar-la, calcular-ne el cost i provar-la.
- Introduir el concepte de recursivitat i la seva aplicació, així com ser capaç d'analitzar la complexitat dels algorismes recursius.

Competències

- Analitzar i resoldre problemes de manera efectiva, i generar propostes innovadores i creatives per aconseguir els objectius.
- Conèixer i utilitzar de manera eficient les tècniques i eines de representació, manipulació, anàlisi i gestió de dades a gran escala.
- Desenvolupar pensament crític per analitzar de manera fonamentada i argumentada alternatives i propostes tant pròpies com alienes.
- Dissenyar, implementar, analitzar i validar solucions algorítmiques eficients i robustes per a problemes computacionals derivats del disseny de sistemes intel·ligents.
- Introduir canvis en els mètodes i els processos de l'àmbit de coneixement per donar respostes innovadores a les necessitats i demandes de la societat.

Resultats d'aprenentatge

1. Analitzar i resoldre problemes de manera efectiva, i generar propostes innovadores i creatives per aconseguir els objectius.
2. Aplicar els principis de la programació orientada a objectes en el desenvolupament de programes.
3. Aplicar estratègies de depuració, prova i correcció per verificar el funcionament correcte d'una solució algorítmica d'acord amb els requisits del problema a resoldre.
4. Avaluar i analitzar la complexitat computacional de les solucions algorítmiques per poder desenvolupar i implementar aquella que garanteixi el millor rendiment.
5. Comprendre les estructures de dades comunes i els algorismes que les utilitzen i manipulen.
6. Conèixer les limitacions i els avantatges de diferents representacions de dades i tenir la capacitat de elegir les estructures més eficients que facilitin l'anàlisi i explotació de les dades posterior.
7. Desenvolupar pensament crític per analitzar de manera fonamentada i argumentada alternatives i propostes tant pròpies com alienes.
8. Desenvolupar programes amb un bon estil de programació i ben documentats.
9. Implementar solucions recursives a problemes de programació.
10. Proposar nous mètodes o solucions alternatives fonamentades.
11. Seleccionar i utilitzar les estructures algorítmiques i de representació de les dades apropiades per a la resolució d'un problema.

Continguts

Estructures de dades dinàmiques:

- Punters, estructures de dades dinàmiques (l·listes, piles, cues).
- Estructures de dades no lineals. Tècniques de hashing. Matrius hash i l·listes hash.
- Recursivitat.
- Algorismes d'ordenació.
- Algorismes de cerca.
- Grafs.
- Arbres.
- Anàlisi de la complexitat dels algorismes.

Programació avançada de diferents estructures de dades dinàmiques en llenguatge C.

Programació bàsica de diferents estructures de dades dinàmiques en llenguatge C++.

Importació de codi C des de Python.

Programació de diferents estructures de dades de Python en C.

Llenguatges de programació:

- Llenguatge estructurat C (bàsic i avançat)
- Llenguatge orientat a objectes C++ (bàsic)
- Llenguatge interpretat Python (coneixements de Fonaments de Programació 1)

Metodologia

Farem servir la plataforma Caronte (<http://caronte.uab.cat/>) per compartir materials, lliurar projectes i altres activitats administratives.

Per matricular-vos a Caronte cal registrar-vos (amb nom, NIU i foto) i matricular-vos a l'assignatura (el codi de l'assignatura es facilitarà el primer dia de classe).

La metodologia docent se centrarà principalment en el treball pràctic. S'hi organitzaran sessions presencials per a la discussió dels continguts teòrics de l'assignatura, seguides d'exercicis i problemes de programació. Més concretament, els diferents tipus d'activitats docents són els següents:

- Classes teòriques: Explicació a pissarra de la part teòrica de cada tema del programa. L'estructura típica d'una classe serà la següent: en primer lloc, es realitzarà una introducció on s'exposaran breument els objectius de la classe i els continguts a tractar. A continuació, s'abordaran els continguts d'un tema, incloent-hi exposicions narratives, desenvolupaments formals que aportin fonaments teòrics i intercalant exemples

pràctics que il·lustrin l'aplicació dels continguts exposats. Finalment, el professorat exposarà les conclusions dels continguts tractats. Les classes teòriques, tot i ser principalment una explicació per part del professorat, seran participatives per als/les alumnes donant-los l'oportunitat de preguntar aquells punts que no acaben d'entendre. Als/les alumnes se'ls plantejaran constantment preguntes i problemes per comprovar la bona comprensió del tema exposat. Hi haurà 2 exàmens per avaluar aquesta part de l'assignatura.

- Classes de resolució de problemes: Resolució conjunta de problemes. Tots els temes aniran acompanyats d'un conjunt de problemes que l'estudiant haurà d'intentar resoldre. En aquest sentit, i a mesura que l'estudiant progressi i aprofundeixi en els coneixements, aquests problemes seran progressivament més complexos. Les classes seran el fòrum natural en què es pugui discutir en comú el desenvolupament del treball pràctic, aportant els coneixements que falten a l'estudiant per tirar-lo endavant. Les classes de resolució de problemes tenen com a missió servir de pont entre les classes teòriques i les classes de laboratori, cosa que fomentarà la capacitat d'anàlisi i síntesi, el desenvolupament del raonament crític i que capacitarà a l'estudiant en la resolució de problemes. Durant el curs, els/les alumnes (en grups de 2 persones) faran diversos exercicis curts i bàsics que seran avaluats. Cada grup resoldrà aquests exercicis bàsics i farà el lliurament en un termini curt corresponent. Aquest exercici busca que els/les alumnes puguin resoldre teòricament els conceptes treballats a les classes teòriques per després poder aplicar-los al cas pràctic de laboratori. Els exercicis estaran disponibles a Caronte.

- Projectes pràctics: els/les estudiants treballaran en parelles i recorreran els diferents passos del cicle de vida del desenvolupament de programari per implementar la solució a un problema. Aquests projectes pràctics completaran les assignatures teòriques on l'estudiant desenvoluparà diversos programes i haurà d'intentar resoldre un problema concret proposat. Aquests projectes hauran de ser lliurats a les dates indicades.

Aquest enfocament de treball s'adreça a promoure l'aprenentatge actiu i desenvolupar competències d'organització i planificació, comunicació oral i escrita, treball en equip i raonament crític. Es valorarà especialment la qualitat dels exercicis realitzats, de la presentació i del funcionament.

Tots els detalls del funcionament, el calendari i les dates d'avaluació s'informaran al començament del curs i apareixeran a la plataforma de comunicació.

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			
Projectes pràctics	18	0,72	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11
Realització d'exercicis bàsics	5	0,2	2, 3, 5, 9, 11
Solucions de problemes	10	0,4	2, 3, 5, 7, 9, 11
Teoria	10	0,4	2, 5, 6, 9, 11
Tipus: Autònomes			
Estudis autònoms	56	2,24	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11
Preparació problemes i exercicis bàsics	10	0,4	2, 3, 4, 5, 6, 9, 11
Preparació projectes	30	1,2	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11

Avaluació

L'objectiu del procés d'avaluació és comprovar que l'alumnat ha assolit els coneixements i les habilitats definits en els objectius de l'assignatura, així com les competències associades.

S'avaluaran tres tipus d'activitats de forma independent, i la suma ponderada en donarà la nota final. Aquestes tres activitats són:

- Teoria (T)
- Solució dels projectes pràctics (PP)
- Resolució d'exercicis bàsics (EXER).

La part de Teoria (T) s'avaluarà amb dos exàmens parcials individuals al llarg del curs. La nota final de Teoria resultarà de la suma ponderada dels dos exàmens ($0,5 \cdot \text{Control 1} + 0,5 \cdot \text{Control 2}$). Hi haurà una segona oportunitat per recuperar aquesta part el dia de juny que tenim assignat. Les parts que no hagin estat superades als exàmens parcials de teoria es podran recuperar per separat. La nota mínima per aprovar aquesta part de Teoria és ≥ 5 . A la recuperació només es podran recuperar els exàmens suspesos, no podent pujar la nota obtinguda (en el cas d'aprovar la part de Teoria). La nota màxima que es pot obtenir en recuperació és un 7.

La part de projectes pràctics (PP) s'avaluarà en grups de 2 persones. Hi ha dues entregues. La nota final sortirà de la suma ponderada de les dues entregues ($0,5 \cdot \text{Lliurament 1} + 0,5 \cdot \text{Lliurament 2}$). Per aprovar el PP la nota mínima haurà de ser ≥ 5 . Només hi ha una oportunitat per aprovar aquesta part (aquesta part no es pot recuperar).

Els exercicis pràctics (EXER) es realitzaran en grups de 2 persones i consistiran a treballar problemes de programació molt concrets, breus i relacionats amb els que es troben a les pràctiques de laboratori. Volem que cada estudiant estudiï la solució d'un conjunt de problemes específics de manera aïllada en el cas més general de la pràctica. El valor d'aquests exercicis és el 20% de la nota final i, atesa la seva naturalesa i finalitat, no són recuperables.

La nota final de l'assignatura serà la suma ponderada de les notes de cadascuna de les tres activitats: 40% de Teoria, 40% de treballs pràctics i 20% de Resolució d'exercicis bàsics. El resultat haurà de ser ≥ 5 .

En cas que un/a estudiant no aprovi l'assignatura per no assolir la nota mínima en alguna de les parts obligatòries (Teoria o Projectes Pràctics), encara que la mitjana ponderada sigui igual o superior a 5, la nota final de l'assignatura serà de 4, 5.

En cas que la mitjana no arribi a 5, la nota oficial serà la nota mitjana obtinguda numèricament.

Si l'estudiant lliura alguna activitat, s'entendrà que participa a la matèria i serà avaluat. Si no lliura cap activitat, llavors es pot considerar No evaluable.

La concessió d'un títol de matrícula d'honor (MH) és una decisió del responsable de la matèria. La normativa de la UAB indica que només es poden concedir MH als/les alumnes que hagin obtingut una nota final igual o superior a 9,00. Es pot atorgar fins al 5% de MH del total d'alumnes matriculats.

Les dates d'avaluació continuada i entrega de treballs estaran publicades a la plataforma de comunicació i podran estar subjectes a possibles canvis per adaptar-se a possibles incidències; Caront informará sempre sobre aquests canvis ja que s'entén que és el mecanisme habitual d'intercanvi d'informació entre professors i alumnes.

Per a cada activitat d'avaluació s'indicarà un lloc, data i hora de revisió en què l'estudiant podrà repassar l'activitat amb el professor. En aquest context, es poden fer reclamacions sobre la qualificació de l'activitat, que serà avaluada pels professors de la matèria. Si l'estudiant no se sotmet a aquesta revisió, aquesta activitat no serà revisada més endavant.

RESUM

Si $(T \geq 5)$ i $(PP \geq 5)$ llavors

$$NF = 0,40 \cdot T + 0,2 \cdot EXER + 0,40 \cdot PP$$

Si (NF >= 5) llavors APROVAT

en cas contrari SUSPÈS

en cas contrari SUSPÈS

Alumnes repetidors: els/les alumnes repetidors que hagin superat pràctiques de laboratori a edicions anteriors de l'assignatura NO PODEN sol·licitar la convalidació d'aquesta part de l'assignatura. Totes les activitats d'avaluació s'han de fer en les mateixes condicions que la resta d'alumnes.

Nota sobre el plagi:

Sense perjudici d'altres mesures disciplinàries que es considerin oportunes, i d'acord amb la normativa acadèmica vigent, les irregularitats comeses per un estudiant que puguin donar lloc a una variació de la qualificació en una activitat avaluable seran qualificades amb zero (0). Les activitats d'avaluació qualificades d'aquesta manera i per aquest procediment no seran recuperables. Si per aprovar l'assignatura cal superar alguna d'aquestes activitats d'avaluació, se suspendrà directament aquesta assignatura, sense possibilitat de recuperar-la en el mateix curs. Aquestes irregularitats inclouen, entre d'altres:

- la còpia total o parcial d'un exercici de laboratori, informe o qualsevol altra activitat d'avaluació;
- deixar que un altre estudiant copii;
- presentar un treball grupal no realitzat íntegrament pels integrants del grup (aplicat a tots els integrants i no només als que no han treballat);
- presentar com a propis materials elaborats per un tercer, encara que es tracti de traduccions o adaptacions, i en general obres amb elements no originals i exclusius de l'alumne;
- disposar de dispositius de comunicació (com ara telèfons mòbils, rellotges intel·ligents, bolígrafs amb càmera, etc.) accessibles durant les proves d'avaluació teòricopràctiques (exàmens individuals);
- parlar amb els companys durant les proves teòricopràctiques individuals (exàmens);
- copiar o intentar copiar altres estudiants durant les proves d'avaluació teòricopràctiques (exàmens);
- utilitzar o intentar utilitzar material escrit relacionat amb la matèria durant les proves d'avaluació teòricopràctiques (exàmens), quan aquestes no hagin estat expressament permeses.
- Si no s'aproveu l'assignatura pel fet que alguna de les activitats d'avaluació no assolix la nota mínima exigida, la nota oficial numèrica serà el valor més baix entre 4,5 i la mitjana ponderada de les notes. Amb les excepcions que s'atorgarà la qualificació de "No Valoratiu" als alumnes que no participin en cap de les activitats d'avaluació, i que la nota oficial numèrica serà el valor més baix entre 3,0 i la nota mitjana ponderada en cas que l'alumne hagi comès irregularitats en un acte d'avaluació (i per tant, el subjecte no pot ser aprovat per compensació). En properes edicions d'aquesta assignatura, l'alumne que hagi comès irregularitats en un acte d'avaluació no serà convalidat de cap de les activitats d'avaluació realitzades.

En resum: copiar, deixar copiar o plagiar (o intentar) a qualsevol de les activitats d'avaluació donarà lloc a un SUSPENS, no indemnitzable i sense convalidacions de parts de l'assignatura en cursos posteriors.

Activitats d'avaluació

Títol	Pes	Hores	ECTS	Resultats d'aprenentatge
1er Control parcial individual	20%	2	0,08	4, 5, 11
2on Control parcial individual	20%	2	0,08	2, 4, 5, 9, 11
Projectes pràctics	40%	2	0,08	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11
Realització d'exercicis bàsics	20%	5	0,2	2, 3, 4, 5, 6, 9, 11

Bibliografia

Mark Allen Weiss: Data Structures and Algorithm Analysis in C (2nd Edition). Pearson, 1996.

Mark Allen Weiss: Data Structures and Algorithm Analysis in C++. 4 th edition. Pearson, 2014.

Kernighan, Brian; Ritchie, Dennis M., The C Programming Language (2nd edition). Englewood Cliffs, NJ: Prentice Hall. 1988.

K.N. King: C Programming: A Modern Approach. 2nd edition. W.W. Norton & Company. 2008

<http://www.cplusplus.com/> : The C++ Resources Network

Programari

Microsoft Visual C++. Visual Studio. Available for UAB students at <https://msdnaa.uab.cat/>

Queue manager (SLURM)

Remote connection to the laboratory (e.g. MobaTex, Eclipse, Visual Studio, etc.). MobaXTerm for Windows [<https://mobaxterm.mobatek.net/>] or Visual Studio Code for all platforms [<https://code.visualstudio.com/>]