

Metodologia de la Programació**2014/2015**

Codi: 102764

Crèdits: 6

Titulació	Tipus	Curs	Semestre
2502441 Enginyeria Informàtica	FB	1	2

Professor de contacte

Nom: Ernest Valveny Llobet

Correu electrònic: Ernest.Valveny@uab.cat

Utilització de llengües

Llengua vehicular majoritària: català (cat)

Grup íntegre en anglès: No

Grup íntegre en català: Sí

Grup íntegre en espanyol: No

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. Tot i això es dedicaran algunes sessions a repassar aquells conceptes introduïts a Fonaments d'Informàtica que siguin més rellevants pel seguiment de l'assignatura.

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 estructures bàsiques de programació introduïts a Fonaments d'Informàtica i completar-los amb la resta de conceptes de programació que han de permetre que al finalitzar el curs l'estudiant tingui una visió completa de la programació imperativa.

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, 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 i arbres binaris), per poder escollir l'estructura de dades més adient per representar la informació associada un problema algorímic.
- 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.
- Conèixer els problemes fonamentals de cerca i classificació. Saber 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).
- Comprendre el concepte de recursivitat i la seva aplicació al disseny d'algorismes, introduint les tècniques de disseny recursiu.

- 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
- 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 la capacitat d'anàlisi, síntesi i prospectiva.
8. Reconèixer i identificar els mètodes, sistemes i tecnologies propis de l'enginyeria informàtica.

Continguts

Tema 1: Introducció a la programació orientada a objectes

- Repàs de les estructures de programació bàsiques: estructures condicionals i iteratives, funcions i procediments, taules, matrius, registres.
- Fitxers. Concepte de fitxer. Fitxers d'accés seqüencial. Operacions amb fitxers.
- Introducció al concepte de classe. Mètodes i atributs. Part privada i pública. Constructors i destructors.

Tema 2: Estructures de dades dinàmiques

- Necessitat d'estructures de dades dinàmiques.
- El concepte d'apuntador. Operacions amb apuntadors.
- Objectes dinàmics.
- Definició, representació i implementació d'estructures de dades dinàmiques: llistes, piles i cues.
- Utilització d'estructures de dades dinàmiques.

Tema 3: Disseny d'algorismes

- Fases en el desenvolupament de l'algorisme. Anàlisi de l'algorisme. Disseny de l'estructura de dades. Disseny arquitectònic.

- Disseny modular i descendent.
- Disseny de proves.

Tema 4: Recursivitat

- Concepte de recursivitat. Aplicació a l'algorísmica. Disseny d'algorismes recursius.
- Aplicació a la cerca i ordenació. Cerca binària. QuickSort
- Tipus de dades recursius. Arbres: definició, representació i implementació. Arbres binaris: representació, recorregut, aplicació a la classificació: HeapSort

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. Així, el principal resultat d'aprenentatge serà la realització d'un projecte de programació que s'haurà d'anar desenvolupant durant tot el curs i que suposarà integrar de forma pràctica gairebé tots els conceptes i eines de programació introduïts a les classes de teoria en la resolució d'un problema real complex. Aquest projecte de programació serà una extensió del projecte desenvolupat al primer semestre a l'assignatura Fonaments d'Informàtica per donar continuïtat i coherència global a les dues assignatures de programació del primer curs.

Apart del projecte de programació, la resolució dels exercicis que es proposaran i discutiran a les sessions de problemes ha de ser també una part important del procés d'aprenentatge de l'alumne, per ajudar-lo a comprendre, integrar i aplicar els conceptes explicats a classe de teoria. Per la seva banda, les sessions de pràctiques de laboratori serviran per aplicar aquests conceptes generals al projecte de programació. A totes les activitats del curs (teoria, problemes i pràctiques) es farà servir el llenguatge de programació C++.

D'aquesta forma, el procés complet d'aprenentatge de l'alumne es fonamentarà en quatre tipus d'activitats diferents: classes de teoria, sessions de problemes, pràctiques de laboratori i el projecte de programació.

Classes de teoria

Les classes de teoria serviran per introduir els conceptes que es detallen al temari de l'assignatura. Les classes es basaran en les explicacions del professor, que s'ajudarà de material complementari en forma de fotocòpies o transparències, i també en una sèrie d'exemples pràctics que es plantejaran als alumnes per facilitar la comprensió i aprenentatge dels conceptes explicats.

L'estudiant haurà de completar les classes de teoria amb el treball personal necessari per acabar d'entendre els continguts de l'assignatura i poder preparar correctament les sessions de problemes. 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.

Sessions de problemes

Les sessions de problemes (anomenades pràctiques d'aula en els horaris) es dedicaran bàsicament a la resolució i discussió de problemes que serveixin d'aplicació dels conceptes explicats a les classes de teoria.

Cada grup de teoria es divideix en dos grups de problemes (grup A i grup B). La divisió dels estudiants entre els dos grups es farà al principi i serà fixa per tot el curs.

Per cada tema es proposarà un petit projecte de programació que l'alumne ha de resoldre de forma autònoma. A les sessions de problemes es treballaran exercicis basats en aquests projectes que els alumnes podran utilitzar per acabar de resoldre el projecte. Al final de cada tema, el projecte s'haurà de lliurar mitjançant el campus virtual (<https://cv.uab.es/>).

Els lliuraments dels exercicis s'haurà de fer en grup. Els grups de treball s'hauran de donar d'alta utilitzant el campus virtual i hauran de ser els mateixos al llarg del curs.

Pràctiques de laboratori

Durant el curs hi haurà 6 sessions de pràctiques de laboratori que es dedicaran principalment al seguiment i avaluació del treball fet per l'alumne en el projecte de programació.

Per cada sessió es proposarà un conjunt d'exercicis relacionats amb el projecte que l'estudiant haurà de fer abans de la sessió. Durant la sessió el professor examinarà el treball realitzat pels estudiants, avaluarà el progrés, corregirà els errors i resoldrà els dubtes que se li plantegin. L'assistència a les sessions de control al laboratori és obligatòria i servirà per determinar la part de la nota d'avaluació continuada relacionada amb les pràctiques.

El treball de les sessions de laboratori s'haurà de fer en grups de 2 persones. Els grups s'hauran de donar d'alta utilitzant el campus virtual i seran els mateixos que pel lliurament del projecte de programació. Per apuntar-se a un dels horaris de les sessions de laboratori s'haurà d'utilitzar el campus virtual.

Pr

El
Fo
i p
Fo

A l

Activitats formatives

Títol	Hores	ECTS	Resultats d'aprenentatge
Tipus: Dirigides			
Classes magistrals	26	1,04	
Explicació i resolució de problemes	13	0,52	
Pràctiques de laboratori	10	0,4	
Tipus: Supervisades			
Seguiment de la resolució del projecte de programació	1	0,04	
Tipus: Autònomes			
Estudi individual	20	0,8	
Implementació del projecte de programació	50	2	
Resolució de problemes de manera individual i grupal	25	1	

Avaluació

L'avaluació de l'assignatura tindrà en compte tres tipus d'activitats d'avaluació:

1. Avaluació continuada: en aquest apartat el lliurament dels exercicis i activitats que es realitzin a les sessions de problemes i de pràctiques.
2. Avaluació individual: en aquest apartat s'inclou el resultat de les proves escrites individuals que es faran al llarg del curs. Hi haurà dues proves parciais 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.
3. Projecte: inclou el treball del projecte de programació. Es divideix en tres lliuraments repartits al llarg del curs.

La nota final de l'assignatura s'obté combinant l'avaluació d'aquestes 3 activitats de la manera següent:

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

$$\text{Avaluació Continuada} = (0.5 * \text{Activitats Problemes}) + (0.5 * \text{Activitats laboratori})$$

$$\text{Projecte} = (0.2 * \text{Entrega Parcial 1}) + (0.2 * \text{Entrega Parcial 2}) + (0.6 * \text{Entrega Final})$$

$$\text{Avaluació Individual} = (0.5 * \text{Parcial 1}) + (0.5 * \text{Parcial 2})$$

No hi ha recuperació de la nota del projecte i de l'avaluació continuada després de la data de lliurament

Condicions per aprovar l'assignatura:

- Nota Final ≥ 5
- Nota Projecte ≥ 4
- Nota Entrega Final Pràctica ≥ 5
- Nota Avaluació Individual ≥ 5
- Nota de cadascun dels parcials ≥ 4

No presentats:

- Un alumne es considerarà no presentat si 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 a la nota de l'expedient de l'alumne.

Convalidacions:

- Pels alumnes repetidors no es convalida cap nota de cap activitat d'avaluació dels anys anteriors

Nota important: còpies i plagis

Sense perjudici d'altres mesures disciplinaries que s'estimin oportunes, i d'acord amb la normativa acadèmica vigent, es qualificaran amb un zero les irregularitats comeses per l'estudiant que puguin conduir a una variació de la qualificació d'un acte d'avaluació. Per tant, copiar o deixar copiar una pràctica o qualsevol altra activitat d'avaluació implicarà suspendre-la amb un zero, i si és necessària superar-la per aprovar, tota l'assignatura quedarà suspesa. No seran recuperables les activitats d'avaluació qualificades d'aquesta forma i per aquest procediment, i per tant l'assignatura serà suspesa directament sense oportunitat de recuperar-la en el mateix curs.

Nota sobre la planificació de les activitats d'avaluació:

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

Activitats d'avaluació

Títol	Pes	Hores	ECTS	Resultats d'aprenentatge
-------	-----	-------	------	--------------------------

Avaluació continuada	30%	0,5	0,02	2, 7, 8
Avaluació individual	40%	4	0,16	1, 5, 6
Projecte de programació	30%	0,5	0,02	1, 3, 4

Bibliografia

1. J. Pujol: Algorismes i Programes. Servei de Publicacions de la UAB, 1996.
2. E. Valveny, R. Benavente, A. Lapedriza, M. Ferrer, J. García: Programació en Llenguatge C. Amb 56 problemes resolts i comentats. Servei publicacions UAB, 2009.
3. M. Jesús Marco Galindo et al. Fonaments de Programació. Publicacions UOC, 2008.
4. L. Joyanes Aguilar : Fundamentos de Programación: Algoritmos, Estructuras de Datos y Objetos. 3ª Edición, Mc. Graw-Hill, 2003.
5. L. Joyanes Aguilar, L. Rodriguez Baena, M. Fernández Azuela: Fundamentos de Programación. Libro de Problemas. 2ª Edición, Mc Graw-Hill, 2003.
6. B.W. Kernighan, D.M. Ritchie: El lenguaje de programación C. 2ª Edición, Prentice Hall, 1986.
7. L. Joyanes, A. Castillo, L. Sánchez, I. Zahonero: Programación en C: libro de problemas, Mc Graw-Hill, 2002.
8. L. Joyanes, I. Zahonero: Programación en C: metodología, estructura de datos y objetos, Mc Graw-Hill, 2001.
9. B.W. Kernighan, R. Pike: La Práctica de la Programación. Pearson Educación, 2000.
10. F. Xhafa, P. Vázquez, J. Marco, X. Molinero, A. Martín: Programación en C++ para ingenieros. Thomson, 2006