

GUIA DOCENT D'INFORMÀTICA

1.- IDENTIFICACIÓ DE L'ASSIGNATURA.

Nom: Informàtica

Codi: 27988

Crèdits: 12

Tipus: Troncal (anual)

2.- OBJECTIUS.

L'objectiu d'aquesta assignatura consisteix a presentar els fonaments bàsics de la programació de computadors digitals entesa com un conjunt de tècniques que permeten escriure programes de computadors de forma correcta i eficient. S'introdueixen els conceptes bàsics de la programació (algorismes, estructures de dades, programes i llenguatges) dins del paradigma de la programació imperativa, incidint especialment en els conceptes de correcció eficiència i generalitat.

3.- CONTINGUTS

1. Introducció

- a. Definicions
- b. Antecedents històrics
- c. Computadors i algorismes: processament de la informació
- d. Llenguatges de programació i programes
- e. Estructura del computador
- f. Codificació de la informació

2. Algorísmica: dades + Control

- a. Definició i descripció d'algorismes
- b. Dades i tipus de dades.
- c. Tipus de dades i variables en C
- d. Operativitat amb les dades
- e. Operadors en C
- f. L'entrada/sortida de la informació

3. Programació estructurada i modular

- a. Estructura general d'un programa i estructures de control
- b. Programació estructurada
- c. Programació modular
- d. Computabilitat i complexitat
- e. Funcions i procediments en C
- f. La directiva #define

4. Tipus d'estructures de dades

- a. Introducció
- b. Vectors
- c. Matrius
- d. Cadenes de caràcters
- e. Registres

5. Tipus de dades abstractes

- a. Definició
- b. Apuntadors i gestió de memòria
- c. Llistes lineals
 - i. Llistes estàtiques
 - ii. Llistes encadenades
 - iii. Llistes doblement encadenades
 - iv. Llistes circulars
 - v. Tipus de llistes especials: Piles i cues

6. Fitxers

- a. Introducció
- b. Definicions i conceptes
- c. Dispositius d'emmagatzemament extern
- d. Organització i accés
- e. Fitxers seqüencials
- f. Fitxers relatius
- g. Fitxers seqüencials indexats

7. Algorismes de cerca i classificació

- a. Introducció
- b. Mètodes de cerca bàsics
 - i. Cerca amb tractament
 - ii. Cerca seqüencial
 - iii. Cerca de cadenes
 - iv. Cerca binària
- c. Mètodes directes de classificació de taules
 - i. Classificació per selecció
 - ii. Classificació per inserció
 - iii. Classificació per intercanvi

8. Algorismes recursius

- a. Definicions i algorismes recursius
- b. Mecanisme de la recursivitat
- c. Aplicacions de la recursivitat
- d. Limitacions de la recursivitat

9. Estructures de dades avançades: grafs i arbres

- a. Definicions
- b. Representació dels grafs en C

- c. Arbres i classes d'arbres
- d. Arbres binaris
- e. Operacions amb arbres

10. Llenguatges de programació

- a. Llenguatges i gramàtiques
- b. Sintaxi i semàntica
- c. Classificació dels llenguatges de programació

4.- TEMPS QUE HA DE DEDICAR UN ALUMNE A L'ASSIGNATURA.

TIPUS D'ACTIVITAT	Descripció	Hores
ACTIVITATS PRESENCIALS	Classes de teoria	52
	Classes de problemes	24
	Classes de pràctiques	34
	Activitats tutoritzades	0
	Realització de proves parcials	0
	Realització d'exàmens finals	6
ACTIVITATS NO PRESENCIALS	Estudi de teoria	35
	Realització de problemes	40
	Preparació de pràctiques	84
	Preparació de treballs	0
	Preparació d'exàmens	25
	TOTAL	300

5.- CAPACITATS O DESTRESES A ADQUIRIR.

Capacitats teòriques.

- Tenir una visió històrica de les necessitats de còmput i les solucions obtingudes
- Diferenciar entre Càlcul i Còmput.
- Tenir una visió àmplia del concepte de programari.
- Entendre l'ordinador com un sistema i les relacions establertes entre les seves parts.
- Relacionar profundament els conceptes de maquinari i programari i les seves dependències.
- Conèixer els passos necessaris des del plantejament d'un problema fins l'obtenció de solucions mitjançant programes informàtics.
- Entendre les limitacions dels sistemes digitals a l'hora de tractar la informació atès el possibles estats de les capacitats elèctriques.
- Valorar la necessitat de codificar la informació per tal de fer-ne el tractament, especialment la codificació per als sistemes informàtics.
- Entendre que el concepte de valor la informació és independent del sistema de representació emprat.

- Conèixer els sistemes de representació numèrica en bases diferents a la decimal, en especial la representació binària, octal i hexadecimal.
- Saber transcriure qualsevol valor enter o fraccionari, positiu o negatiu, als sistemes de representació binària més emprats als ordinadors.
- Entendre la relació entre dígit emprats a una representació i el rang i precisió que poden prendre les representacions.
- Entendre els mecanismes generals que permeten codificar qualsevol mena d'informació en format digital.
- Dominar el concepte d'algorisme amb independència de la seva representació.
- Conèixer els diferents tipus de dades simples i les possibilitats que ofereixen els llenguatges de programació per emmagatzemar-les.
- Conèixer l'operativitat possible amb cada tipus de dada simple i les limitacions que tenen associades.
- Conèixer les estructures de control de flux dels programes: seqüència, selecció i iteració.
- Entendre la programació modular com el primer paradigma per programar correctament.
- Saber les diferències entre els diferents tipus de mòduls que integren els programes i la importància de les interfícies de comunicació entre els mòduls.
- Saber diferenciar els conceptes de computabilitat dels problemes i complexitat dels algorismes
- Conèixer les estructures de dades homogènies i heterogènies, així com les seves aplicacions.
- Entendre les estructures de dades abstractes i la seva utilitat, en particular les llistes encadenades.
- Conèixer els mecanismes de magatzematge permanent d'informació i els suports que ho permeten.
- Entendre les principals organitzacions i arquitectures dels sistemes de fitxers.
- Conèixer les operacions estàndard amb fitxers.
- Conèixer els principals algorismes de cerca i classificació de la informació i la complexitat que tenen associada per tal de poder triar l'algorisme més adequat a cada situació.
- Entendre el mecanisme de la recursivitat i el seu funcionament dins de la màquina.
- Introduir-se en l'ús d'estructures de dades dinàmiques multidimensionals (grafs i arbres). Entendre la seva utilitat per representar entitats reals complexes per descripció o relacions.

Capacitats de problemes

- Saber transcriure qualsevol valor enter o fraccionari, positiu o negatiu, als sistemes de representació binària més emprats als ordinadors.
- Saber realitzar operacions aritmètiques simples emprant els sistemes de representació binària habituals als ordinadors.
- Saber representar algorismes mitjançant redactats sintètics, diagrames de flux i pseudocodi.
- Saber dissenyar algorismes per resoldre problemes coneguts (especialment matemàtics) i representar-los de manera que sigui fàcil la seva transcripció a un programa de C.
- Saber calcular la complexitat de diferents algorismes.
- Saber representar entitats complexes mitjançant grafs o arbres.
- Poder aplicar algorismes eficients de càlcul en grafs (cerca de recorreguts)

- Poder aplicar algorismes eficaços de gestió d'arbres (recorregut, cerca, inserció i eliminació de nodes)

Capacitats de pràctiques

- Instal·lar entorns de programació per a Windows o consola.
- Obrir, modificar, gravar, compilar i executar programes.
- Saber localitzar els errors sintàctics mitjançant les eines que facilita l'entorn de programació.
- Conèixer els diferents tipus de blocs conceptuals de què consten els programes en C.
- Conèixer i emprar les funcions d'entrada/sortida estandard en C.
- Conèixer els tipus de dades simples que facilita C, les propietats característiques de cadascuna i els operadors per treballar amb elles.
- Conèixer les principals biblioteques de funcions de C estàndard
- Emprar les estructures de control de flux dels programes amb les possibles variants que ofereix C.
- Realitzar el disseny top down per extraure funcions i procediments per generar programes d'abast mitjà.
- Conèixer l'utilitat de funcions i procediments dissenyats pel programador.
- Dominar correctament el disseny d'interfícies de comunicació entre mòduls.
- Saber emprar correctament els operadors de direcció i els apuntadors.
- Definir estructures de dades repetitives i emprar-les correctament.
- Definir estructures de dades heterogènies per tractar entitats complexes.
- Saber emprar estructures de dades mixtes: registres/vectors
- Realitzar una gestió correcta de la memòria en temps d'execució: ús de les funcions de memòria associades a apuntadors.
- Saber crear, modificar, gravar i destruir fitxers de dades des del llenguatge de programació i tenir cura dels possibles errors que generen els dispositius on s'emmagatzema la informació.
- Saber programar els principals algorismes de cerca i classificació per treballar amb dades emmagatzemades en arrays.

6.- REQUISITS.

Per tal que un alumne pugui superar l'assignatura és imprescindible que tingui un domini adequat dels mecanismes de representació de la informació per tal de poder triar els tipus de dades més adequats per a cada entitat que es vulgui representar en els programes. Haurà de tenir força destresa en l'ús dels mecanismes de control de flux dels programes i de la programació modular, sabent diferenciar quins subproblemes han de solucionar-se mitjançant procediments i quins mitjançant funcions.

Tanmateix, és imprescindible compassar els avenços en els coneixements teòrics amb el desenvolupament de pràctiques per aplicar-los. Atès que el temps requerit per codificar els algorismes pot arribar a ser molt elevat, molts dels problemes plantejats es resoldran com d'algorisme amb representació a un nivell adequat, sense arribar a cercar programes executables.

Des del punt de vista pràctic l'alumne haurà de saber crear programes aplicant el mecanisme de disseny top-down, codificant adequadament els mòduls del programa, tenint cura que els programes resultin robustos als errors associats amb les entrades de dades per part de l'usuari. Tanmateix s'hauran de redactar manuals d'usuari dels programes i fer-ne una descripció abstracta de l'estructura i funcionament dels mateixos. Es valorarà la feina en equip, únic mecanisme que permet la construcció de programes de gran entitat i l'ús i disseny de biblioteques de funcions.

7.- METODOLOGIA.

L'assignatura disposa, al llarg del curs acadèmic, de dues hores setmanals de teoria, una de problemes i tres de pràctiques, hores, aquestes darreres, que es distribuïran en dues hores quinzenals per cada alumne més una hora setmanal dedicada a consultes; totes elles desenvolupades al laboratori docent de matemàtiques de manera que pot dedicar-se a la construcció o depuració dels programes que hagi de presentar.

Està fortament recomanat que l'estudiant assisteixi a totes les sessions de teoria i problemes així com a les de pràctiques que el correspongui.

Al campus virtual, a l'adreça ccd.uab.es se subministra tots els materials i informacions necessaris pel desenvolupament de l'assignatura.

Pel que fa a les classes de teoria, l'alumne disposarà, abans del començament de cada capítol de l'assignatura, d'uns apunts que li poden servir d'orientació. És molt convenient que l'estudiant hagi treballat personalment sobre aquest material abans d'anar a classe.

Periòdicament, l'estudiant rebrà unes llistes de problemes que ha de pensar i sobre els quals es treballarà a les classes de problemes.

Tanmateix, juntament amb les llistes de problemes per revisar a classe, es lliurarà una curta llista de problemes per tal que els realitzi i lliuri resolts en les dates en què s'indiqui. Com a poc un cop per semestre, cada alumne haurà de presentar-se a una entrevista per tal d'explicar com ha fet els problemes que ha lliurat. El resultat d'aquestes entrevistes tindrà una importància d'un 10% de la qualificació de teoria de l'assignatura.

Per tal de facilitar la realització d'aquest problemes, els alumnes podran comptar amb el suport del professor al laboratori docent a l'hora de consultes de pràctiques.

8.- AVALUACIÓ.

Dins el termini marcat pel professor, l'estudiant haurà de lliurar els problemes resolts (llistes curtes) i presentar-se a les entrevistes quan així se'ls requereixi. La qualificació mitjana de les entrevistes (**Ent**) equivaldrà al 10% de la qualificació de teoria.

El dia 3 de Febrer de 2005 realitzarem el primer parcial de l'assignatura. Denotem per **E1** a la qualificació obtinguda en aquest examen. El segon examen parcial es realitzarà el dia 16 de juny, i la qualificació la denotarem per **E2**.

La qualificació de Teoria (**T**) s'obtindrà mitjançant al següent equació:

$$T = 0.1 * Ent + 0.45 * E1 + 0.45 * E2$$

En el cas que un estudiant obtingui mitjançant els exàmens parcials una qualificació superior o igual a **5** es considerarà que té la Teoria aprovada. En cas contrari, haurà de presentar-se a la prova final del 30 de juny en què haurà d'examinar-se de les parts que tingui suspeses.

Pel que fa a la qualificació de pràctiques (**P**) s'obtindrà a partir de la qualificació ponderada de les cinc pràctiques en C a desenvolupar, segons la següent fórmula:

$$P = (P1 + P2) * 0.15 + (P3 + P4) * 0.2 + P5 * 0.3$$

En el cas que un estudiant no s'hagi presentat als dos examens parcials y hagi deixat de lliurar una pràctica, la qualificació a juny serà de **No presentat**.

Per obtenir una qualificació positiva al final de l'assignatura, cal superar la teoria i la pràctica, en cas contrari, a setembre l'estudiant s'haurà d'examinar de la teoria si és aquesta part la que té suspesa, o bé presentar una pràctica d'estiu si no ha superat la part pràctica.

La qualificació global respondrà a la fórmula:

$$F = 0.75 * T + 0.25 * P$$

Nota important: Tot i que aquestes dates d'examen són oficials, poden patir alguna variació.

Es recomana, per tant, que un mes abans de la realització de cada prova es demani la confirmació de les dates.

9.- BIBLIOGRAFIA.

Bibliografia bàsica

1. Antonakos, JL, Mansfield, KC. *Programación estructurada en C* Prentice Hall. Madrid 1997
Guia molt completa i didàctica del llenguatge C Ansi, que emprarem a les pràctiques
2. J. Pujol, *Algorismes i programes*. Col.lecció Materials n. 27. Servei de Publicacions de la UAB, 1996, 1998.
Conté un enfocament didàctic que se segueix de prop a l'assignatura

3. N. Wirth. *Algoritmos+Estructuras de datos=Programas*. Castillo, 1980.
Clàssic sobre la teoria de la programació estructurada, tipus de dades i algorismes d'ús general

Bibliografia complementaria

4. L. Goyanes. *Fundamentos de programación. Algoritmos y estructuras de datos* McGraw-Hill 1989
5. G. Dromey *How to solve it by computer*. Prentice Hall, 1982
6. A. Prieto i altres *Introducción a la informática*. McGraw-Hill, 1989

10.- PROFESSORAT.

Jaume Gil Martínez
Teoria, problemes i pràctiques

QC/2010 (ETSE)
Jaume.Gil@uab.es

935813572