



Universitat de Lleida

# GUIA DOCENT

# **PROCESSADORS DE**

# **LLENGUATGE**

Coordinació: Assignatura que s'imparteix durant el 2n semestre del 4t curs de la titulació Grau en Enginyeria Informàtica. Correspon al mòdul d'especialització en computació.

Coordinadora de l'assignatura Tere Alsinet

Any acadèmic 2013-14

## Informació general de l'assignatura

<b>Denominació</b>	PROCESSADORS DE LENGUATGE
<b>Codi</b>	102043
<b>Semestre d'impartició</b>	2n Q Avaluació Continuada
<b>Caràcter</b>	Obligatòria
<b>Nombre de crèdits ECTS</b>	9
<b>Crèdits teòrics</b>	0
<b>Crèdits pràctics</b>	0
<b>Coordinació</b>	Assignatura que s'imparteix durant el 2n semestre del 4t curs de la titulació Grau en Enginyeria Informàtica. Correspon al mòdul d'especialització en computació.  Coordinadora de l'assignatura Tere Alsinet
<b>Departament/s</b>	Informàtica i Enginyeria Industrial
<b>Informació important sobre tractament de dades</b>	Consulteu <a href="#">aquest enllaç</a> per a més informació.
<b>Distribució de crèdits</b>	Jordi Planes Cid 7 Maria Teresa Alsinet Bernadó 2
<b>Horari de tutoria/lloc</b>	Per fixar l'hora de les tutories enviar un correu electrònic al professor.

Jordi Planes Cid  
Maria Teresa Alsinet Bernadó

## Objectius acadèmics de l'assignatura

Els objectius d'aprenentatge de l'assignatura són per una banda conèixer les etapes, tècniques i algorismes que intervenen en el procés de traducció dels llenguatges de programació i utilitzar eines de suport al disseny i implementació de cada etapa.

A més dels llenguatges imperatius clàssics analitzarem les característiques principals i les tècniques d'implementació associades a altres llenguatges com els llenguatges lògics, funcionals, de scripting, orientats a objectes, i distribuïts i concurrents.

## Competències

### Competències estratègiques de la Universitat de Lleida

- Domini de les Tecnologies de la Informació i la comunicació.
- Domini d'una llengua estrangera.

### Competències transversals de la titulació

- Capacitat d'anàlisi i síntesi.

### Competències específiques de la titulació

- Capacitat per tenir un coneixement profund dels principis fonamentals i models de la computació i saber-los aplicar per interpretar, seleccionar, valorar, modelar, i crear nous conceptes, teories, usos i desenvolupaments tecnològics relacionats amb la informàtica.
- Capacitat per avaluar la complexitat computacional d'un problema, conèixer estratègies algorítmiques que puguin conduir a la seva resolució i recomanar, desenvolupar i implementar la que garanteixi el millor rendiment d'acord amb els requisits establerts.
- Capacitat per conèixer els fonaments teòrics des llenguatges de programació i les tècniques de processament lèxic, sintàctic i semàntic associades, i saber aplicar-les per a la creació, disseny i processament dels llenguatges.

## Continguts fonamentals de l'assignatura

Estructura de l'assignatura en temes:

- 1- Introducció als llenguatges de programació i a les tècniques de traducció
- 2- Anàlisi lèxica.
- 3- FLex (The Fast Lexical Analyzer )
- 4 - Anàlisi sintàctica: Analitzadors sintàctics descendents i ascendents.
- 5- Yacc (Yet Another Compiler Compiler)
- 6- Traducció dirigida per la sintaxi

7- Taules de símbols

8- SymTab

9- Comprovació de tipus

10- Representacions intermèdies i generació de codi intermedi

11- Gestió de memòria en l'entorn d'execució

12- Optimització de codi

13- Generació de codi objecte

## Eixos metodològics de l'assignatura

Les classes de l'assignatura s'estructuren en 4 hores setmanals de resolució de problemes pràctics al laboratori i 2 hores setmanals de caràcter més expositiu on es presentaran els algorismes, tècniques i eines de traducció de cada etapa del procés de traducció.

Els estudiants resoldran exercicis pràctics durant les sessions de laboratori i abordaran la preparació i presentació d'un llenguatge de programació no clàssic a la resta de grups.

## Pla de desenvolupament de l'assignatura

El temari de l'assignatura s'estructura en dues parts.

La primera aborda l'especificació i reconeixement dels components lèxics dels llenguatges de programació, estudia les tècniques d'anàlisi sintàctica i mostra com integrar les rutines semàntiques amb els algorismes d'anàlisi sintàctica. La formació de l'estudiant es complementa amb l'estudi d'eines especialitzades de suport al disseny i implementació de components específics dels sistemes de traducció. Sota aquest marc presentem l'eina d'especificació i reconeixement de llenguatges JFLAP, l'eina de generació d'analitzadors lèxics flex, de generació d'analitzadors sintàctics ascendents yacc i de gestió de taules de símbols SymTab. A més, presentarem eines com ANTLR (ANother Tool for Language Recognition) que permeten la generació d'analitzadors sintàctics ascendents.

La segona part de l'assignatura aborda les fases d'anàlisi semàntica, optimització de codi i generació de codi objecte. Mostrem com incorporar al procés d'anàlisi rutines semàntiques que permetin la gestió d'àmbits, la comprovació de tipus, la generació de codi intermedi per a les principals construccions dels llenguatges imperatius i l'assignació de memòria. S'estudien optimitzacions de codi dependents de la representació intermèdia i de la màquina objecte, i com generar codi per a una màquina objecte.

Per completar la formació dels estudiants en les etapes de generació de codi objecte i d'optimització, durant la setmana del 5 de maig, comptarem amb el suport d'en Paulo Matos, Enginyer Informàtic amb una àmplia experiència en el disseny i implementació de generadors de codi per a dispositius físics. Paulo Matos forma part de l'equip d'enginyers de l'empresa BroadCom i forma part de l'equip de desenvolupadors GNU GCC. Més concretament els aspectes que ens presentarà són:

- 1.- GNU GCC internals.
- 2.- The Register Allocation Problem
- 3.- Source code optimization in Systems on Chip

Finalment, a més dels llenguatges imperatius clàssics analitzarem les característiques principals i les tècniques d'implementació associades a altres llenguatges com els llenguatges lògics, funcionals, de scripting, orientats a objectes, i distribuïts i concurrents. Per al seu estudi i anàlisi, els estudiants escolliran un llenguatge el qual presentaran a la resta del grup.

## Sistema d'avaluació

La nota final serà la suma de les següents notes:

- Resolució d'una col·lecció de problemes d'especificació i reconeixement de llenguatges emprant l'eina JFLAP : 10 % de la nota
- Nota de pràctiques : 60 % de la nota
  - Pràctica 1 : ús de l'eina de generació automàtica d'analitzadors lèxics lex. 10 % de la nota.
  - Pràctica 2: ús de l'eina de generació automàtica d'analitzadors sintàctics yacc. 15 % de la nota.
  - Pràctica 3: ús de l'eina d'implantació de taules de símbols i integració amb les eines lex i yacc. Es proposa desenvolupar un traductor a codi intermedi (3-adreces) per a un llenguatge imperatiu reduït. 35 % de la nota.
- Treball en grup de dues o tres persones : 30 % de la nota
  - Treball: Anàlisi de les característiques pròpies d'un llenguatge de programació no clàssic. Aspectes d'implementació. Treball a realitzar en grup de dues o tres persones. Presentació oral a la resta de grup.

## Bibliografia i recursos d'informació

A més del material disponible a l'apartat de Recursos, podeu consultar les referències següents:

Referències

- [1] A.V. Aho, M. Lam, R. Sethi, and J.D. Ullman. Compilers: Principles, Techniques, and Tools. Addison-Wesley Series in Computer Science, Reading, Massachusetts. Second Edition. 2006.
- [2] Dick Grune, Henri E. Bal, Criel J. H. Jacobs, Koen G. Langendoen. Modern Compiler Design. Jonh Wiley and Sons, England, 2000.
- [3] Andrew W. Appel, Maia Ginsburg. Modern Compiler Implementation in C. Cambridge University Press, 1998.
- [4] John Levine. Flex & bison: Text Processing Tools. O'Reilly, 2009.
- [5] [Reinhard Wilhelm](#), Helmut Seidl, [Sebastian Hack](#): Compiler Design - Syntactic and Semantic Analysis. Springer 2013.
- [6] Helmut Seidl, [Reinhard Wilhelm](#), [Sebastian Hack](#): Compiler Design - Analysis and Transformation. Springer 2012.
- [7] [Reinhard Wilhelm](#), Helmut Seidl: Compiler Design - Virtual Machines. Springer 2010

Eines de generació en C/C++:

- Flex: <http://flex.sourceforge.net/>
- Yacc: <http://www.gnu.org/software/bison/>

Eina especificació de llenguatges:

- JFLAP: <http://www.jflap.org/jflaptmp/>

Eines de generació en Java:

- JFlex: <http://jflex.de/>
- Cup: <http://www2.cs.tum.edu/projects/cup/>
- Ant: <http://ant.apache.org/>
- ANTLR: <http://www.antlr.org/>