

## Guia docent

### Identificació de l'assignatura

<b>Assignatura / Grup</b>	21742 - Compiladors I / 1
<b>Titulació</b>	Grau d'Enginyeria Informàtica (Pla 2014) - Tercer curs Grau d'Enginyeria Informàtica (Pla 2010) - Tercer curs
<b>Crèdits</b>	6
<b>Període d'impartició</b>	Primer semestre
<b>Idioma d'impartició</b>	Català

### Professors

Professor/a	Horari d'atenció als alumnes					
	Hora d'inici	Hora de fi	Dia	Data d'inici	Data de fi	Despatx / Edifici
Pedro Antonio Palmer Rodríguez ( <i>Responsable</i> ) <a href="mailto:pere.palmer@uib.es">pere.palmer@uib.es</a>	16:30	17:30	Divendres	10/09/2018	25/01/2019	Anselm Turmeda D142
	10:30	11:30	Divendres	10/09/2018	28/06/2019	Anselm Turmeda D142

### Contextualització

Els compiladors són programes que prenen com a dades codi escrit en llenguatges de programació d'alt nivell i el converteixen a versions equivalents que siguin fàcilment executables per una màquina. L'assignatura té per objectiu explicar les dificultats d'aquest procés i les tècniques que existeixen per a resoldre-les. Aquestes darreres fan ús combinat del bagatge adquirit en diferents assignatures prèvies, sobretot, teoria d'autòmats, estructures de dades, llenguatges de programació i estructura de computadors.

El coneixement de l'estructura i funcionament d'un compilador aporta diferents avantatges a l'estudiant i al professional:

En primer lloc articula i dona sentit a molts dels coneixements que haurà vist en assignatures anteriors (teoria d'autòmats, estructures de dades, arquitectura de computadors, teoria de grafs, etc.) i hi trobarà la manera de combinar-los per resoldre un problema de tanta importància pràctica com és la traducció d'un llenguatge de programació en codi màquina.

En segon lloc, per un programador, la comprensió de com un compilador tractarà el codi que ell escriu l'ajudarà a ser conscient de les implicacions que tenen les seves decisions en l'eficiència del codi i l'ajudaran a construir programes de major qualitat.

Finalment, la pràctica de l'assignatura constitueix un excel·lent exercici de programació en el qual l'estudiant s'enfronta per primera vegada a la dificultat d'escriure un programa gran.

En l'assignatura de Compiladors I s'aborden els aspectes frontals d'un compilador, és a dir els que depenen del llenguatge font: l'anàlisi lèxica, l'anàlisi sintàctica i la comprovació de tipus. En l'assignatura de Compiladors II s'aborden els aspectes dorsals: generació de codi intermedi, optimització de codi, gestió de memòria i generació de codi assemblador.

Acabats el dos cursos, l'estudiant estarà en condicions de construir un compilador i d'entendre els detalls d'implementació de les construccions pròpies dels llenguatges d'alt nivell. Les tècniques que haurà conegut

## Guia docent

serveixen també per a la construcció de formatejadors de text (LaTeX i similars), llenguatges gràfics (Postscript, GIF, JPEG), i llenguatges de descripció de hardware (VDHL, Verilog).

### Requisits

---

L'estudiant que cursi aquesta assignatura haurà de tenir sòlids coneixements de programació, estructures de dades, teoria d'autòmats, llenguatges de programació i estructura de computadors.

#### Essencials

- 1) Programació I
- 2) Programació II
- 3) Algorítmia
- 4) Estructures de dades
- 5) Teoria de la computació
- 6) Estructura de computadors I
- 7) Estructura de computadors II

#### Recomanables

Seria recomanable que haguessin cursat l'assignatura Llenguatges de programació.

### Competències

---

#### Específiques

- \* CI302 Capacitat per conèixer els fonaments teòrics dels llenguatges de programació i les tècniques de processament lèxic, sintàctic i semàntic associades, i saber aplicar-les per a la creació, el disseny i el processament de llenguatges. .

#### Genèriques

- \* CTR01 Capacitat d'anàlisi i síntesi, d'organització, de planificació i de presa de decisions. .
- \* CTR02 Capacitat d'anàlisi crítica i de proposta i aplicació de noves solucions. .
- \* CTR03 Capacitat per adquirir de forma autònoma nous coneixements. .
- \* CTR04 Capacitat per a la recerca de recursos i de gestió de la informació en l'àmbit de la informàtica. .
- \* CTR07 Capacitat per comunicar conceptes propis de la informàtica de manera oral i escrita en diferents àmbits d'actuació. .

## Guia docent

### Bàsiques

\* Podeu consultar les competències bàsiques que l'estudiant ha d'haver assolit en acabar el grau a l'adreça següent: [http://estudis.uib.cat/ca/grau/comp\\_basiques/](http://estudis.uib.cat/ca/grau/comp_basiques/)

## Continguts

### Continguts temàtics

#### TEMA I. Estructura General

##### TEMA I. VISIÓ GENERAL DEL PROCÉS DE COMPILACIÓ

1. Funcions d'un compilador. Història. Tipus de llenguatge font. Tipus de llenguatge objecte. Intèrprets. La compilació i el disseny de llenguatges de programació. La compilació i el disseny de llenguatges màquina. Les tècniques de compilació i llur aplicació en altres àmbits: processadors de text, interfícies d'usuari, eines d'ajuda a la programació. Compilació incremental. La importància de la qualitat en els diagnòstics d'error.

2. Fases d'un compilador. Anàlisi lèxica. Anàlisi sintàctica. Comprovació de tipus. Generació de codi intermedi. Optimització de codi. Generació de codi executable. Resultats de cada fase. Taules de símbols. Fases i passos.

#### TEMA II. Anàlisi lèxica

##### TEMA II. ANÀLISI LÈXICA

3. Unitats lèxiques. Models, valors, atributs i lexemes. Les expressions regulars com a mitjà de descripció de models. Llur abast i adequació. Llur insuficiència en llenguatges de programació antics. Extensions per a tractar aquests casos. Interrelació entre el disseny del lèxic d'un llenguatge i el disseny de la seva sintaxi. Descripció de lèxics. Ambigüitats i criteris de resolució.

4. Autòmats finits i reconeixement lèxic. Expressions regulars i autòmats no deterministes. Determinització d'autòmats. Minimització d'autòmats. Descripció d'un lèxic i autòmat finit associat. Diferències entre un autòmat finit i un analitzador lèxic. Anticipació. Algorisme de reconeixement. La seva adaptació al cas d'expressions regulars exteses. Anticipació múltiple i les seves repercussions en l'eficiència de l'anàlisi. El tractament de les paraules reservades.

5. Eines de generació d'analitzadors lèxics. Principis. Llenguatges i metallenguatges. Algorismes de determinització i minimització. Tècniques de compactació de les taules que representen la funció de transició.

6. Taula de noms. Registre i caracterització d'identificadors i literals. Estructures de dades eficients.

7. Tractament i recuperació d'errors. Tècniques bàsiques. Modus pànic. Recuperació associada a l'estat de l'autòmat. Interrelació entre nivells del compilador.

#### TEMA III. Anàlisi sintàctica

##### TEMA III. ANÀLISI SINTÀCTICA

8. Descripció de la sintaxi. Gramàtiques incontextuals. Abast i adequació. Símbols terminals i unitats lèxiques. Arbres d'anàlisi. Derivacions per la dreta i per l'esquerra. Gramàtiques ambigües. El determinisme i la seva repercusió en l'eficiència de l'anàlisi.

9. Analitzadors descendents recursius. Esquema. Requisits d'aplicació: gramàtiques LL(k) i LL(1).

## Guia docent

10. Gramàtiques LL(1). Algoritmes d'identificació. Conjunts  $First(a)$  i  $Follow(A)$ . Llur construcció. Transformacions de gramàtiques conduents a llur conversió en LL(1): eliminació de la recursivitat esquerra i factorització. Inclusió estricta dels llenguatges que admeten gramàtiques LL(1) respecte als incontextuals deterministes.

11. Analitzadors descendents iteratius. Construcció automàtica de llurs taules de transició. L'algoritme d'anàlisi descendent i el seu invariant.

12. Anàlisi ascendent. Principis: reduccions i desplaçaments. L'algoritme d'anàlisi LR i el seu invariant. Altres tipus d'anàlisi ascendent.

13. Anàlisi LR simple. Principis. Items LR simples. Estats i tancament simple. Algoritme de construcció. Conflictes desplaçament-reducció i reducció-reducció. Inclusió estricta dels llenguatges que admeten gramàtiques LR simples i el dels llenguatges incontextuals deterministes.

14. Anàlisi LR canònica. Principis. Items LR canònics. Estats i tancament canònics. Algoritme de construcció. Identitat entre els llenguatges que admeten gramàtiques LR canòniques i els llenguatges incontextuals deterministes. El problema del volum de les taules LR canòniques.

15. Anàlisi LALR. Principis. Inclusions estrictes entre els llenguatges que admeten gramàtiques LR simples, LALR i LR canòniques. Algoritme eficient de construcció de taules LALR.

16. Anàlisi LR en gramàtiques ambigües. Resolució extragramatical de les ambigüitats. Conveniència en relació amb la simplicitat de la descripció del llenguatge. Conveniència en relació a l'eficiència de l'analitzador sintàctic.

17. Eines de generació d'analitzadors LR. Llenguatge i metallenguatge. Diagnòstic de conflictes. Tècniques de compactació de les taules que representen les funcions de transició.

18. Tractament i recuperació d'errors. Tècniques bàsiques. Modus pànic. Recuperació associada a l'estat de l'autòmat. Casos ascendent i descendent.

19. Comparació de gramàtiques LL(1) i LALR(1). Simplicitat. Generalitat. Recuperació d'errors. Volum de les taules. Vincle amb l'anàlisi semàntica.

### TEMA IV. Anàlisi semàntica I: Eines

#### TEMA IV. ANÀLISI SEMÀNTICA I: EINES

20. Gramàtiques d'atributs. Definició. Atributs heretats i sintetitzats. Avaluació d'atributs: cas general. Gramàtiques S-atribuïdes i l'avaluació de llurs atributs. Gramàtiques L-atribuïdes i l'avaluació de llurs atributs. Marcadors. Traducció dirigida per la sintaxi. Anàlisi sintàctica i semàntica en un sol pas.

21. Taules de símbols. Funcions de la taula de símbols. Dependències respecte a les regles de visibilitat del llenguatge. Dependències respecte a l'estructura de passos del compilador. Tècniques d'implementació eficient en llenguatges de blocs. Representació d'índexos de vectors, camps de tuples, paràmetres formals de subprogrames. Sobrecàrrega de símbols. Importació i exportació de declaracions en blocs.

### TEMA V. Anàlisi semàntica II: compr. de tipus

#### TEMA V. ANÀLISI SEMÀNTICA II: COMPROVACIÓ DE TIPUS

22. Fonaments de la comprovació de tipus. Objectius. Sistemes de tipus. Comprovació estàtica i dinàmica. Representació de tipus.

23. Processat de declaracions. Atributs i rutines semàntiques per al tractament de declaracions. Declaracions de tipus. Declaracions de variables. Declaracions de subprogrames. Gramàtiques S-atribuïdes per al tractament de matrius, tuples i subprogrames. Àmbits declaratius.

## Guia docent

24. Processat de sentències. Atributs i rutines semàntiques per al tractament d'expressions. Gramàtiques S-atribuïdes per sistemes de referenciació que inclouen matrius, tuples i invocació de subprogrames. Processat d'assignacions, repeticions i alternances.

25. Comprovació de tipus en llenguatges orientats a objectes: sobrecàrrega i polimorfisme. Algoritmes de filtrat de possibilitats en llenguatges amb sobrecàrrega de símbols i operadors. L'implicació en l'estructura de passos del tractament d'expressions. La comprovació de tipus en llenguatges que admeten tipus polimòrfics. Variables de tipus. Algoritme d'unificació. Processat del dispatching.

## Metodologia docent

### Activitats de treball presencial (2,4 crèdits, 60 hores)

Modalitat	Nom	Tip. agr.	Descripció	Hores
Classes teòriques	Classes teòriques i de problemes	Grup gran (G)	S'hi explica la manera de resoldre els diferents aspectes del procés de compilació que es tracten en el temari i part de la teoria que hi dona suport. Només part, perquè bona part dels aspectes teòrics ja s'han explicat en assignatures precedents.  Un cop desenvolupats els aspectes teòrics, es dediquen unes setmanes de classes a la resolució magistral de problemes que els estudiants haurien d'haver intentat resoldre prèviament pel seu compte.	60

A començament del semestre hi haurà a disposició dels estudiants el cronograma de l'assignatura a través de la plataforma UIBdigital. Aquest cronograma inclourà almenys les dates en què es faran les proves d'avaluació contínua i les dates de lliurament dels treballs. A més, el professor o la professora informará els estudiants si el pla de treball de l'assignatura es durà a terme a través del cronograma o per una altra via, inclosa la plataforma Aula Digital.

### Activitats de treball no presencial (3,6 crèdits, 90 hores)

Modalitat	Nom	Descripció	Hores
Estudi i treball autònom individual o en grup	Pràctica	Els estudiants, en grups petits de entre dos i quatre membres, hauràn de desenvolupar un compilador. Durant l'assignatura Compiladors I, n'hauran de resoldre la part frontal (anàlisi lèxica, sintàctica, la taula de noms i la taula de símbols).  En l'assignatura de Compiladors II reprendran el treball efectuat fins a arribar a la generació de codi ensamblador.  A mitjans del quadrimestre hi haurà dues sessions de control destinades al seguiment de la feina efectuada i, sobretot, de correcció a temps de decisions que puguin perjudicar el desenvolupament posterior del treball.	90

## Guia docent

### Riscs específics i mesures de protecció

Les activitats d'aprenentatge d'aquesta assignatura no comporten riscos específics per a la seguretat i salut dels alumnes i, per tant, no cal adoptar mesures de protecció especials.

### Avaluació de l'aprenentatge dels estudiants

#### Frau en elements d'avaluació

D'acord amb l'article 33 del Reglament acadèmic, "amb independència del procediment disciplinari que es pugui seguir contra l'estudiant infractor, la realització demostradorament fraudulenta d'alguns dels elements d'avaluació inclosos en guies docents de les assignatures comportarà, a criteri del professor, una menysvaloració en la seva qualificació que pot suposar la qualificació de «suspens 0» a l'avaluació anual de l'assignatura".

#### Classes teòriques i de problemes

Modalitat	Classes teòriques
Tècnica	Proves objectives ( <b>recuperable</b> )
Descripció	S'hi explica la manera de resoldre els diferents aspectes del procés de compilació que es tracten en el temari i part de la teoria que hi dona suport. Només part, perquè bona part dels aspectes teòrics ja s'han explicat en assignatures precedents. Un cop desenvolupats els aspectes teòrics, es dediquen unes setmanes de classes a la resolució magistral de problemes que els estudiants haurien d'haver intentat resoldre prèviament pel seu compte.
Criteris d'avaluació	És imprescindible aprovar l'examen per fer mitja. Competències: CI302, CTR01, CTR02

Percentatge de la qualificació final: 50%

#### Pràctica

Modalitat	Estudi i treball autònom individual o en grup
Tècnica	Proves orals ( <b>recuperable</b> )
Descripció	Els estudiants, en grups petits de entre dos i quatre membres, hauràn de desenvolupar un compilador. Durant l'assignatura Compiladors I, n'hauran de resoldre la part frontal (anàlisi lèxica, sintàctica, la taula de noms i la taula de símbols). En l'assignatura de Compiladors II reprendran el treball efectuat fins a arribar a la generació de codi ensamblador. A mitjans del quadrimestre hi haurà dues sessions de control destinades al seguiment de la feina efectuada i, sobretot, de correcció a temps de decisions que puguin perjudicar el desenvolupament posterior del treball.
Criteris d'avaluació	Competències: CI302,CTR01,CTR02, CTR03,CTR04,CTR07

Percentatge de la qualificació final: 50%

### Recursos, bibliografia i documentació complementària

## Guia docent

### Bibliografia bàsica

---

- Aho, A. V., M. S. Lam, R. Sethi, and J. D. Ullman, *Compilers: Principles, Techniques and Tools*, Addison-Wesley, 2006. Sens cap mena de dubte, les edicions precedents d'aquest llibre han estat el text més universalment usat a les universitats per aquesta assignatura. El tractament de la part frontal del compilador és excel·lent. En canvi, la part d'optimització i generació de codi ensamblador queda descrita de manera molt més imprecisa. Tampoc tracta la generació de codi per a llenguatges funcionals.
- Wirth, N., *Compiler Construction*, Addison-Wesley, 1996. És un llibre breu i divulgatiu. Molt apropiat per a una introducció ràpida, però clarament insuficient com a suport del curs.
- Wilhelm, R. and D. Maurer, *Compiler Design*, Addison-Wesley, 1995. Discuteix de manera molt equilibrada els problemes de la compilació per als diferents tipus de llenguatges (imperatius, funcionals, lògics). Ho intenta també per als diferents tipus d'arquitectures (CISC, RISC, paral·leles), però se'n surt de manera molt més superficial.
- Levine, J. R., A. Mason and D. Brown, *Lex and Yacc*, O'Reilly, 1992. Descriu en detall aquestes dues eines per a la generació d'analitzadors lèxics i sintàctics, respectivament.

### Bibliografia complementària

---

- Aho, A. and J. D. Ullman, *The Theory of Parsing, Translation and Compiling. Vol. I: Parsing*, Prentice-Hall, 1972, vol. II: *Compiling*, Prentice-Hall, 1973. Llibre ja antic però encara plenament vigent. Conté tot el fonament matemàtic en el que es basen els algorismes emprats pels textos de la bibliografia bàsica.

### Altres recursos

---

Uns programes per a la generació automàtica d'analitzadors lèxics i sintàctics que poden ser descarregats lliurement des d'una pàgina Web de la UIB.

