

4. Model de dades relacional
  - 4.1. Estructura de dades relacionals
  - 4.2. Regles d'integritat
  - 4.3. Àlgebra relacional
  - 4.4. Llenguatge SQL
  - 4.5. Optimització de consultes
  - 4.6. Normalització i disseny
5. Altres models de bases de dades
  - 5.1. Model jeràrquic
  - 5.2. Model en xarxa
  - 5.3. Model de llistes invertides
6. Entorn d'un sistema de bases de dades
  - 6.1. Concurrència
  - 6.2. Recuperacions
  - 6.3. Control d'integritats
7. Sistemes distribuïts
8. Bases de dades orientades a objectes
9. Bases de dades documentals

### Pràctiques

Les pràctiques es realitzaran utilitzant els sistemes RDB i Oracle sobre MS-DOS.

### Bibliografia

- DATE, C.J.: *Introduction to database systems*. Vol. I, 5<sup>a</sup> ed. Addison Wesley. 1986.
- DATE, C.J.: *Introduction to database systems*. Vol II, Addison Wesley, 1986.
- MARTIN, J.: *Organización de las bases de datos*. Prentice-Hall.
- ULLMAN, J.D.: *Principles of databases systems*. Computer Science Press. 1980.

### 21289 - COMPILADORS

#### Contingut

1. Introducció. Conceptes bàsics. Generalitats.
2. Anàlisi lexicogràfica: escàners. Anàlisi lexicogràfica: consideracions pràctiques.
3. Gramàtiques i anàlisi sintàctica: Parser. Gramàtiques lliures del context. Notació. Característiques de les gramàtiques i llenguatges de programació. Gramàtiques i *parsers* LL(1).

4. Anàlisi semàntica. Traducció dirigida per la sintaxi. Taula de símbols. Anàlisi semàntica de declaracions. Anàlisi semàntica d'expressions.
5. Organització de memòria en temps d'execució. Ambient d'execució i assignació estàtica de memòria. Assignació de la pila. Assignació dinàmica de memòria.
6. Generació de codi. Generació de codi d'expressions. Processar referències a estructures de dades.

### Pràctiques

Realització d'un compilador/intèrpret simple.

### Bibliografia

- TREMBLAY, J.P.; SORENSON, P.G.: *The Theory and Practice of Compiler Writing*. McGraw-Hill.
- FISHER, Ch.N.; LEBLANC jr, R.J.: *Crafting a Compiler with C*. The Benjamin/Cumming Publishing Company, Inc.
- AHO, A.V.; SETHI, R.; ULLMAN, J.D.: *Compiladores: principios, técnicas y herramientas*. Addison-Wesley.
- WATT, D.A.: *Programming Language Processors*. Prentice Hall. 1993

### 21290 - ENGINYERIA DEL SOFTWARE I

#### Objectius

Un primer objectiu és donar una visió global i ordenada del procés de desenvolupament del *software* que no sigui simplement el de la programació –que constitueix, només, una fase dins de tot el procés de la enginyeria del *software*.

En aquesta primera part, ens centrem en les dues primeres etapes del cicle de la vida del *software*: anàlisi i disseny. Estudiarem dos tipus de metodologies: orientat al flux de dades i orientat a l'objecte.

L'objectiu es que l'alumne sigui capaç de utilitzar aquestes metodologies per tal de realitzar l'anàlisi i el disseny de problemes reals, que li permetrà abordar la creació de *software* de manera rigorosa.

#### Contingut

- I. Introducció. Definició. Evolució i crisi del *software*. Cicle de vida. Paradigmes del *software*.

2. Fonaments de l'anàlisi de requeriments. Introducció. Tasca d'anàlisi. Problemàtica associada. Reconeixement del problema. Principis d'anàlisi i especificació. Document d'especificació: definició, principis i representació.
3. Anàlisi orientada al flux de dades o anàlisi estructurada. Anàlisi estructurada. Notació: diagrama de flux de dades. Mecànica de l'anàlisi estructurada.
4. Introducció metodologia orientat a l'objecte. Abstracció. Encapsulació. Modularitat. Jerarquia. Tipatge. Conceptes: classe, atributs, mètode, objecte i missatge.
5. Anàlisi orientada a l'objecte. Introducció a la metodologia Coad i Yourdon. Identificació de classes-objectes. Definir atributs. Identificar estructures. Definir serveis. Identificar temes.
6. Fonaments del disseny del *software*. Procés de disseny. Fonaments del disseny: modularitat, abstracció, refinament. Disseny de dades, arquitectònic, procedural. Documentació del disseny.
7. Disseny orientat al flux de dades o disseny estructurat. Procés del disseny. Anàlisi de transformació i de transacció. Heurístiques de disseny. Postprocesament i optimització del disseny.
8. Disseny orientat a l'objecte. Metodologia de Coad i Yourdon. Disseny del domini del problema. Disseny del domini de la interfície. Disseny del gestor de dades.
9. Temes addicionals I. Anàlisi orientada a l'estructura de dades. Diagrama de Warnier. Desenvolupament de sistemes estructurats de dades. Sistema de desenvolupament de Jackson.
10. Temes addicionals II. Disseny orientat a l'estructura de dades. Disseny i estructures de dades. Desenvolupament de sistemes Jackson. Desenvolupament de sistemes estructurats de dades. Derivació de l'estructura lògica de sortida i del procés. Lògica de processos complexos.
11. Temes addicionals III. Mètodes formals d'anàlisi. Estat actual. Característiques d'un llenguatge d'especificació formal. Especificació formal en Z.

### Pràctiques

Hi haurà dues pràctiques, una d'anàlisi i disseny orientat al flux de dades, i l'altra de disseny orientat a l'objecte, que es desglossaran en quatre entregues. Es faran en grups de dues persones i en un format que especificarà el professor de pràctiques.

### Bibliografia

#### Bibliografia de consulta

– PRESSMAN, R.S.: *Ingeniería del software, un enfoque práctico*. McGraw-Hill (3a. edició), 1993.

- MYNATT, B.T.: *Software engineering with student project guidance*. Prentice-Hall, 1990
- COAD, P.; YOURDON, E.: *Object- Oriented Analysis*. Yourdon Press, 1991
- COAD, P.; YOURDON, E.: *Object- Oriented Desing*. Yourdon Press, 1991
- YOURDON, E.: *Análisis estructurado moderno*. Prentice-Hall, 1993

#### Bibliografia adicional

- BOOCH, G.: *Object oriented desing with applications*. The Benjamin/Cummings Publishing Company. 1990.
- DEMARCO, T.: *Structured analysis and system specification*. Yourdon Press. 1979.
- SOMERVILLE, I.: *Sotware engineering*. 3a. i 4a. Ed. Addison-Wesley. 1992

## 21291 - ENGINYERIA DEL SOFTWARE II

### Objectius

En aquesta assignatura s'estudien les fases del cicle de vida del *software* que no s'han vist a Enginyeria I. L'objectiu continua sent que l'estudiant aprengui a desenvolupar *software* amb una metodologia rigorosa, i que faci servir tècniques d'enginyeria en les fases de desenvolupament que s'expliquen ara. A part d'això, es posa l'accent en dos temes: el disseny d'interfícies gràfiques d'usuari en el marc del sistema de finestres X, i l'enginyeria del *software* assistida per ordinador (CASE).

### Contingut

1. Disseny d'interfícies d'usuari. Factors de qualitat de la IU. Tipus d'usuaris i estils d'interacció. Missatges de sistema i d'error.
2. Interfícies gràfiques d'usuari Sistema de finestres X. Visió general d'X Window System. Avantatges d'X. El servidor X. Comunicació entre clients i servidor. Clients programes d'aplicació. Recursos.
3. Programació d'aplicacions X Window amb la *toolkit Xt Intrinsics*. *Look and Feel*, *Toolkits i Widgets*. Model de programació dirigida per *events*. Funcions bàsiques d'*Xt Intrinsics*. Funcions gestores d'*events* i *callback*. El conjunt de *widgets* d'OLIT. Comunicació entre clients.
4. Codificació. Factors en l'elecció d'un llenguatge de programació. Gestió d'excepcions. Excepcions en C i C++. Compilació separada. *Make*. Adopció d'un estil de programació (C, C++).
5. Tècniques de prova. Concepte de prova del *software*. Proves de caixa blanca: camí bàsic, prova de les estructures de control. Proves de caixa negra: participació equivalent, anàlisi de valors límit. Eines CASE de prova.