

950  
36

UNIVERSIDAD DE BUENOS AIRES  
FACULTAD DE CIENCIAS EXACTAS Y NATURALES

- 1. DEPARTAMENTO: Computación
- 2. CUATRIMESTRE: Primero de 1995.
- 3. ASIGNATURA: PROGRAMACION PARALELA Y DISTRIBUIDA.
- 4. CARRERA: Licenciatura en Ciencias de la Computación
- 5. CARACTER DE LA MATERIA: Optativa.
- 6. NUMERO DE CODIGO DE CARRERA: 18
- 7. NUMERO DE CODIGO DE MATERIA: 578.
- 8. PUNTAJE: 2 (plan '93), 3 (plan '87).
- 9. PLAN DE ESTUDIOS AÑO: 1993 y 1987.
- 10. DURACION DE LA MATRERIA: Cuatrimestral
- 11. HORAS DE CLASE SEMANAL:
  - a) TEORICAS 3 HS.
  - b) LABORATORIO 3 HS.
  - c) PROBLEMAS
  - d) SEMINARIOS
- 12. CARGA HORARIA TOTAL: 6 HORAS
- 13. ASIGNATURAS CORRELATIVAS: Sistemas operativos y algoritmos y estructura de datos III (plan '93).  
Matemática discreta y arquitectura y sistemas operativos (plan '87).
- 14. FORMA DE EVALUACION: Examen Final
- 15. PROGRAMA Y BIBLIOGRAFIA: Adjuntas a esta hoja

FECHA:

N.S.  
-----  
Firma del Profesor

-----  
Firma del Director

Lic. Nelson Serejer  
-----  
Aclaración de la Firma

APROBADO POR RESOLUCION CD 1456/95

Lic. ROBERTO BEVILACQUA  
DIRECTOR ADJUNTO INTERINO  
DEPARTAMENTO DE COMPUTACION  
-----  
Sello Aclaratorio

**Programación Paralela y Distribuida**  
**Programa para el primer cuatrimestre de 1995**  
**Prof. : Nelson Sprejer**

Teóricas: 3 hs. semanales  
Prácticas: 3 hs. semanales clase + 3 hs. semanales consultas.

**I. Introducción:** La *cooperación* como modelo de solución eficiente de problemas. Modelos de paralelismo. Clasificaciones. Conceptos básicos. Tipos de propiedades. Introducción a especificación y verificación de programas paralelos: Lógica Temporal, Redes de Petri. Ejemplos de problemas característicos de ambientes paralelos/distribuidos.

**II. Programación en MIMD:**

**a. El modelo de programación paralela.** UNITY. La filosofía detrás de UNITY. Sintaxis y semántica de un programa UNITY. Supuestos: fairness, punto fijo, liveness. Ejemplos básicos. La lógica de UNITY. Ejemplos avanzados.

**b. El modelo de programación distribuida.** CSP. La filosofía detrás de CSP. Sintaxis y semántica de un programa CSP. Ejemplos básicos. La lógica de CSP. Supuestos: fairness, liveness, DTC. Ejemplos avanzados.

**III. Programación en SIMD:** Máquinas PRAM. La filosofía detrás del modelo PRAM. Variantes del modelo PRAM. Técnicas básicas. Algoritmos en grafos. Sorting. SIMD sobre redes de interconexión. Conceptos básicos. Simulación de máquina PRAM.

**IV. Distintos lenguajes para programación MIMD/SIMD:** Concurrent C, OCCAM, Ada, Argus; Linda, Fortran 90. Algunas palabras sobre programación funcional y lógica concurrente. Características diferenciales y comparaciones.

**V. Algoritmos y protocolos:** Presentación de problemas típicos de entornos distribuidos: Detección de terminación. Instantánea global de un sistema distribuido. Detección de deadlock. Coordinación de comités.

**VI. Miscelánea:** Medidas de eficiencia y complejidad de algoritmos. Decidibilidad y computabilidad. Más sobre especificación y verificación formal. El tiempo en sistemas distribuidos. Metodologías y nuevas construcciones de lenguajes. Relación entre lenguajes de programación y hardware. Programación paralela vs. compiladores paralelizantes.

**Bibliografía Básica**

1. T. Bräunl, *Parallel Programming - An introduction*, Prentice-Hall 1993.
2. M. Ben-Ari *Principles of Concurrent and Distributed Programming* C.A.R. Hoare Series Editor, Prentice-Hall 1990.
3. K. Many Chandy, Joydev Misra *Parallel Program Design - A foundation*, Addison-Wesley 1988.
4. Michel Reynal *Distributed Algorithms and Protocols* John Wiley and Sons. 1988.
5. C.A.R. Hoare *Communicating Sequential Processes* Prentice-Hall 1985.
6. David Harel *Algorithmics - The spirit of computing (2nd. edition)* Addison-Wesley 1992.
7. J. JaJa - *An introduction to Parallel Algorithms* Addison-Wesley 1992.
8. H. Bal et al. - *Programming Languages for Distributed Computing Systems* ACM Comp.Surveys 9/89.