

Asignatura: **Programación Concurrente**  
Titulación/es: Ingeniería Informática.  
Créditos: 6  
Carácter: Obligatoria  
Curso: 3º  
Temporalidad: Cuatrimestral (Segundo Cuatrimestre)  
Departamento: Informática  
Profesor: Julián Ruiz Fernández

Prerrequisitos: Metodología y Tecnología de la Programación (1º)  
Lógica (1º)  
Estructuras de Datos y de la Información (2º)  
Ampliación de Programación (2º)  
Sistemas Operativos (2º)

Correquisitos: Ampliación de Sistemas Operativos (2º)

### **Objetivos:**

Proporcionar al alumno la capacidad de diseñar programas concurrentes y paralelos. Estudiamos los problemas que plantea el diseño de programas concurrentes así como los conceptos y mecanismos que se han propuestos para su diseño. Estudiamos, también, algunos lenguajes utilizados en programación concurrente.

### **Docencia:**

3 horas semanales de teoría y problemas.  
1 hora semanal de prácticas de laboratorio.

### **Evaluación:**

Se realizará un examen final de la asignatura que constará de una parte relativa a las prácticas de laboratorio y otra de teoría y problemas, debiendo aprobar ambas por separado para superar la asignatura.

## **1. Introducción**

- 1.1. Objetivos.
- 1.2. Desafíos.
- 1.3. Definición
- 1.4. Evolución
- 1.5. Ejemplo
- 1.6. Definiciones
- 1.7. Semánticas
- 1.8. Motivación.
- 1.9. Problemas
- 1.10. Arquitecturas

## **2. Especificación de la Concurrencia.**

- 2.1. Ejecución Concurrente
- 2.2. Procesos
- 2.3. Concurrencia en el lenguaje
- 2.4. Características de los procesos
- 2.5. Jerarquía de Procesos
- 2.6. Representación de Procesos.
- 2.7. Implementación de Procesos.

## **3. El Problema de la Exclusión Mutua.**

- 3.1. Definición
- 3.2. Esquema General
- 3.3. Propiedades a satisfacer
- 3.4. Soluciones
- 3.5. Algoritmo de Dekker
- 3.6. Algoritmo de Peterson
- 3.7. Algoritmo de la Panadería.

## **4. Semáforos.**

- 4.1. Problemas Resolubles con semáforos
- 4.2. Semáforos Binarios
- 4.3. Semáforos Generales
- 4.4. Tipos de Semáforos
- 4.5. Aplicaciones:
  - 4.5.1. Sincronización
  - 4.5.2. Garantizar la Exclusión Mutua
  - 4.5.3. Gestión correcta de Recursos Compartidos
- 4.6. Implementaciones
- 4.7. Problema del Productor-Consumidor
- 4.8. Problema del Barbero Dormilón

## **5. Regiones Críticas Condicionales.**

- 5.1. Regiones Críticas.
- 5.2. Ventajas e Inconvenientes.
- 5.3. Aplicaciones.

## **6. Monitores.**

- 6.1. Introducción.
- 6.2. Esquema
- 6.3. Ejemplo
- 6.4. Condiciones
- 6.5. Ventajas e Inconvenientes.
- 6.6. Regiones Críticas Condicionales vs. Monitores
- 6.7. Operaciones Adicionales
- 6.8. Semántica de Signal
- 6.9. Ejemplos.

## **7. Comunicación Directa.**

- 7.1. Modelos.
  - 7.1.1. Paso Asíncrono de Mensajes.
  - 7.1.2. Paso Síncrono de Mensajes.
  - 7.1.3. Llamadas a Procedimientos Remotos (RPC)
  - 7.1.4. Citas (Rendez-vous).
- 7.2. Tareas en Ada
- 7.3. Entradas. Semántica de accept.
- 7.4. Diferencias entre llamadas a Procedimientos y llamadas a Entradas.
- 7.5. Ejemplos.
  - 7.5.1. Exclusión Mutua.
  - 7.5.2. Sincronización.
  - 7.5.3. Simulación de Semáforos Binarios.
- 7.6. Comunicaciones Simétricas y Asimétricas.
- 7.7. Elección entre varias alternativas: Instrucción Select.
  - 7.7.1. Semántica.
  - 7.7.2. Ejemplos.
  - 7.7.3. Alternativa terminate.
  - 7.7.4. Alternativa else.
  - 7.7.5. Alternativa delay.
- 7.8. Evaluación de las citas: Consideraciones.
- 7.9. Ejemplos.

## **8. Activación y Terminación**

- 8.1. Modelos de Activación.
- 8.2. Estados de un Proceso.
- 8.3. Terminación de Procesos.

## 9. Programación Concurrente en Java.

- 9.1. Hilos en Java
- 9.2. Programación con hilos: Lanzamiento y Control.
- 9.3. Exclusión Mutua.
- 9.4. Sincronización
- 9.5. Problema del Productor-Consumidor.
- 9.6. Terminación.
- 9.7. Entrada/Salida entre hilos.

## BIBLIOGRAFÍA BÁSICA.

ANDREWS, G. R. *Foundations of Multithreaded, Parallel, and Distributed Programming*. Addison Wesley, 2000. ISBN 0-201-35752-6

BARNES, J. *Programming in Ada 95*. 2<sup>nd</sup> Ed., Addison Wesley, 1998. ISBN 0-201-34293-6

BEN-ARI, M. *Principles of Concurrent and Distributed Programming*. Prentice Hall, 1990. ISBN 0-13-711821-X

BURNS, A., WELLINGS, A. *Concurrency in ADA*. Cambridge University Press, 1999. ISBN 0-521-62911-X

LEA, D. *Concurrent Programming in Java, Design Principles and Patterns* 2<sup>nd</sup> Ed. Addison-Wesley, 2000. ISBN 0-201-31009-0. Edición en español: *Programación Concurrente en Java, Principios y Patrones de Diseño* Ed. Addison-Wesley, 2000. ISBN 84-7829-038-9

## BIBLIOGRAFÍA COMPLEMENTARIA.

HARTLEY, S. J. *Concurrent Programming. The Java Programming Language*. Oxford University Press, 1998. ISBN 0-19-511315-2

PALMA, J.T., GARRIDO, M.C., SANCHEZ, F., QUESADA, A. *Programación Concurrente* Thomson-Paraninfo, 2003. ISBN 84-9732-184-7

PÉREZ MARTÍNEZ, J. *Programación Concurrente*. Ed. Rueda, 1990. ISBN 84-7207-059-X