

Lección 1: Introducción a la Programación de Sistemas Concurrentes y Distribuidos

- De la programación secuencial a la concurrente y distribuida
- Programas secuenciales, concurrentes y distribuidos
- La asignatura de PSCD
 - organización
 - bibliografía recomendada
 - contenidos del curso
 - notación algorítmica

Temas
**LAS NUEVAS
TECNOLOGÍAS
LIDERAN LA
RECUPERACIÓN
DE LA
ECONOMÍA**

La demanda de ingenieros y sus sueldos se disparan en Barcelona



• 'Start-up' y multinacionales atraen a informáticos y 'telecos' de todo el mundo

ROSA SALVADOR,
BARCELONA
08/01/2017 02:00
Actualizado a
09/01/2017 15:54



Los sueldos se aceleran, pero aún no han llegado al nivel precrisis y son inferiores a los de la UE (Ana Jiménez)

[ELIGE TU CARRERA >](#)

Las carreras más demandadas en un mundo cambiante

Estudios sobre empresa, tecnología y salud despuntan en un entorno de constante cambio



SERGIO C. FANJUL

23 JUN 2019 - 00:47 CEST

También se vive un auge en titulaciones como la Ingeniería Informática o la **Ingeniería Industrial**. "Se requieren perfiles que se adapten a las nuevas

2.3.2 Informática y Tecnologías de la Información y la Comunicación

La valoración de los egresados en el ámbito de la Informática y TIC sitúa en el podio a las siguientes universidades: la **Universidad Politécnica de Valencia**, que consolida el liderato, seguida por la **Universidad de Zaragoza**, que escala de la novena a la segunda posición, y por la **Universidad Politécnica de Cataluña**, que pierde una posición respecto a 2016 y dos respecto a 2015, cuando se situaba a la cabeza en este ámbito.

INFORMÁTICA Y TIC	ADAPTACIÓN COMPETENCIAS Y NECESIDADES
Universidad Politécnica de Valencia	8,04
Universidad de Zaragoza	7,95
Universidad Politécnica de Cataluña	7,85
Universidad de Sevilla	7,80
Universidad Carlos III	7,59
Universidad de Barcelona	7,58
Universidad Complutense de Madrid	7,56
Universidad del País Vasco	7,38
Universidad de Oviedo	7,09
Universidad Politécnica de Madrid	6,95

RANKING
**UNIVERSIDAD
EMPRESA**
FUNDACIÓN **EVERIS**

III RANKING UNIVERSIDAD-EMPRESA

De la programación secuencial a la concurrente y distribuida

- Caso 1:

```
integer x := 0
--x = 0
x := 1
--x = 1
```

```
integer x := 0
--x = 0
x := 2
--x = 2
```



```
integer x := 0
--x = 0
x := 1 || x := 2
--x = ?????
```

De la programación secuencial a la concurrente y distribuida

- Caso 2:

```
integer x := 0
--x = 0
x := x + 1
--x = 1
```

```
integer x := 0
--x = 0
x := x + 1
--x = 1
```



```
integer x := 0
--x = 0
x := x + 1 || x := x + 1
--x = ?????
```

Programas secuenciales, concurrentes y distribuidos

- Un programa concurrente se compone de *procesos y objetos compartidos*
- Un proceso es un *programa secuencial* ejecutado en algún procesador
- Los objetos compartidos se construyen o bien mediante *memoria compartida* o bien mediante una *red de comunicación*

Programas secuenciales, concurrentes y distribuidos

- Para cooperar, los procesos deben comunicar
 - la comunicación permite a un proceso influir en la ejecución de otro
- La comunicación puede tener retrasos
 - lo que implica que la información obtenida por un proceso respecto a otro puede estar desfasada
- En consecuencia: **desarrollar programas concurrentes/distribuidos correctos es más difícil que diseñar un conjunto de procesos correctos**

La asignatura de PSCD

- **“Since the development of CS2001, some relevant trends in the evolution of the discipline of computer science have become apparent. These include”**
 - *The emergence of security as a major area of concern*
 - *The growing relevance of concurrency*
 - *The pervasive nature of net-centric computing*

Computer Science Curriculum 2008:
An Interim Revision of CS 2001

Report from the Interim Review Task Force

includes update of the CS2001 body of knowledge plus commentary

December 2008

Association for Computing Machinery

IEEE Computer Society

La asignatura de PSCD

- Dónde encaja en nuestro plan de estudios

PLAN DE ESTUDIOS - GRADO EN INGEN		
Cuatrimestre 1º	Tipo	Créd.
INTRODUCCIÓN A LOS COMPUTADORES	FB	6
FUNDAMENTOS DE ADMINISTRACIÓN DE EMPRESAS	FB	6
MATEMÁTICAS 1	FB	6
MATEMÁTICAS 2	FB	6
PROGRAMACIÓN 1	FB	6
Cuatrimestre 3º	Tipo	Créd.
TEORÍA DE LA COMPUTACIÓN	FB	6
SISTEMAS OPERATIVOS	OB	6
REDES DE COMPUTADORES	OB	6
PROGRAMACIÓN DE SISTEMAS CONCURRENTES Y DISTRIBUIDOS	OB	6
ESTRUCTURA DE DATOS Y ALGORITMOS	OB	6



La asignatura de PSCD. Resultados del aprendizaje

- El estudiante terminará con un **conocimiento** profundo de cuáles son las características específicas de los sistemas concurrentes y distribuidos
- Conocerá los **problemas** generados por el acceso concurrente a datos y recursos, así como las **soluciones conceptuales y tecnológicas** que se han dado a los mismos
- Tendrá nociones de qué son los sistemas **tiempo real**, y sistemas **basados en eventos**
- Conocerá **herramientas** para el diseño y programación de programas con características concurrentes y/o distribuidas

La asignatura de PSCD. Trabajo del estudiante

- 60 horas de actividades presenciales
 - sesiones de teoría, problemas y prácticas de laboratorio
- 30 horas de trabajo en grupo
- 55 horas de trabajo y estudio individual efectivo
 - estudio de textos, resolución de problemas, preparación de clases y prácticas, desarrollo de programas, etc.
- 5 horas dedicadas a distintas pruebas de evaluación

La asignatura de PSCD. Evaluación

- **Actividades de evaluación**

- Prueba práctica en el laboratorio: 15%
- Realización y defensa de trabajos prácticos en grupo: 15%
- Prueba escrita: 70%

- **Calificación final**

- media ponderada de las tres partes
- necesario llegar a 4.0 puntos sobre 10.0 en las prácticas de laboratorio y 5.0 puntos sobre 10.0 en el trabajo en grupo y la prueba escrita
- en caso de no alcanzar ese mínimo en alguna de las partes
 - mínimo entre 4.0 y el resultado de ponderar con los porcentajes de cada parte.

La asignatura de PSCD. Evaluación

- **Convocatoria de Septiembre:** consta de 2 pruebas:
 - Prueba escrita (70%)
 - Prueba de Laboratorio (30%)
 - Esta última prueba será obligatoria para todos aquellos alumnos que no hayan superado la Prueba práctica en el laboratorio y/o el Trabajo en grupo
 - Sólo se guardan para septiembre las notas de Prácticas y del Trabajo en grupo si ambas partes cumplen con el requisito de la nota mínima

La asignatura de PSCD. Profesorado

- **Pedro Álvarez**

- **alvaper@unizar.es**, despacho 2.16, ext. 5541
- tutorías: **<http://diis.unizar.es/ConsultaTutorias.php>**

- **Joaquín Ezpeleta**

- **ezpeleta@unizar.es**, despacho 1.17, ext. 1955
- tutorías: **<http://diis.unizar.es/ConsultaTutorias.php>**

Bibliografía recomendada

- M. Ben-Ari
Principles of Concurrent and Distributed Programming
Addison-Wesley, 2006

- A. Williams
C++, Concurrency in Action
Manning, 2012

- M.L. Liu
Computación distribuida. Fundamentos y aplicaciones
Ed. Pearson- Addison Wesley, 2004.

<http://webdiis.unizar.es/asignaturas/pscd/doku.php>

Contenidos del curso

1. Introducción a la programación concurrente
 2. Modelado y análisis de sistemas concurrentes
 3. El problema de la sección crítica
 4. Semáforos
 5. Monitores
 6. Introducción a la programación distribuida
 7. Comunicación síncrona y asíncrona en entornos distribuidos
 8. Diseño y programación de sistemas distribuidos en Java
 9. Algoritmos distribuidos
 10. Introducción a la programación dirigida por eventos y tiempo real
-

Contenidos del curso. Trabajo de laboratorio

1. La programación concurrente. Threads y datos compartidos. Problemas de interferencias
2. Solución al problema de la sección crítica
3. Programación con semáforos
4. Programación con monitores
5. Programación de sistemas distribuidos
6. Análisis del proyecto en grupo

Notación algorítmica

```
constant integer i := 27
integer v := 27
real r
constant real PI := 3.1415926535 --el de siempre
boolean ha_ido_bien

integer array[1..100] mis_datos := (1..100, 0)

type integer array[1..n,1..n] mat_cuad
type integer array[1..n] vect
mat_cuad A
vect b,x
```

```
-----  
process multiplica
```

```
  mat_cuad A
```

```
  vect b,x
```

```
-----  
  operation obtener_mat(REF mat_cuad m)
```

```
  --Pre: TRUE
```

```
  --Post: m se ha leído de la entrada estándar
```

```
-----  
  operation obtener_vect(REF vect v)
```

```
  --Pre: TRUE
```

```
  --Post: v se ha leído de la entrada estándar
```

```
  ....
```

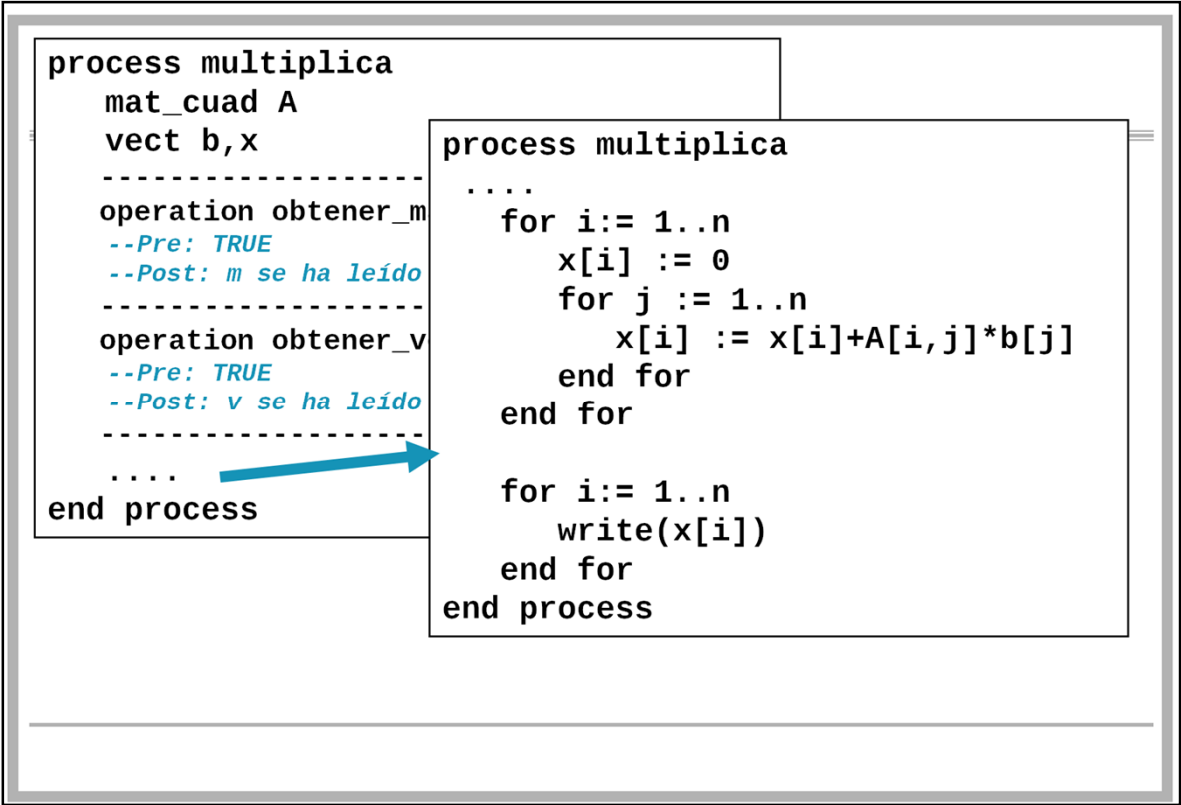
```
end process
```

```
end for
```

```
end process
```

```
.n
```

```
x[i]+A[i,j]*b[j]
```



```
operation obtener_mat(REF mat_cuad m)
--Pre: TRUE
--Post: m se ha leído de la entrada estándar
```

```
    for i:= 1..n
      for j := 1..n
        read(m[i,j])
      end for
    end for
end operation
```

```
operation obtener_vect(REF vect v)
--Pre: TRUE
--Post: v se ha leído de la entrada estándar
integer i := 1
while i<=n
  read(v[i])
  i := i+1
end while
end operation
```