

Tema7. Comunicación y sincronización entre procesos



Contenidos

1. Concurrencia y recursos compartidos
2. Acceso exclusivo y secciones críticas.
3. Mecanismos de comunicación y sincronización.
4. Comunicación mediante paso de mensajes.
5. Llamadas al sistema relacionadas con la comunicación entre procesos.
6. Modelo de gestión de recursos: esquema cliente-servidor.
7. Ejemplos de gestores de recursos.

Bibliografía:

- [Capítulo 9] F.M. Márquez: UNIX. Programación Avanzada 3ª Edición. Rama, 2004.
- [Apartados 5.2 y 5.5] C. Rodríguez, I. Alegria, J. González, A. Lafuente: Descripción Funcional de los Sistemas Operativos. Síntesis, 1994
- [Capítulo 5] W. Stallings: Sistemas Operativos. 5º Ed. Pearson Prentice-Hall, 2005.
- [Capítulo 8 y 9.3] G. Nutt: Sistemas Operativos. 3º Ed. Pearson Addison Wesley, 2001
- [Capítulos 7 y 8] A. Silberschartz: Operating Systems Concepts. Sixth, John Wiley & Son, 2003

7.1 Concurrencia y Recursos Compartidos

S.O. Multiprogramado:

mas de un proceso ejecutándose concurrentemente o en paralelo (si hay más de una CPU)

□ Procesos Independientes.

■ Ejecución determinista:

la ejecución del mismo programa con los mismos datos produce los mismos resultados

□ Procesos Dependientes:

■ Modelo productor-consumidor

■ Cliente-Servidor

Spooling de la impresora

Concurrencia y Recursos Compartidos

- ❑ **Procesos Dependientes:**

Procesos que **cooperan** o compiten.

- Ejecución no determinista:**

La ejecución de los mismos programas con los mismos datos no siempre produce los mismos resultados.

El resultado de la ejecución depende del orden de ejecución de los procesos.

- ❑ Ejecución ORDENADA de los procesos.

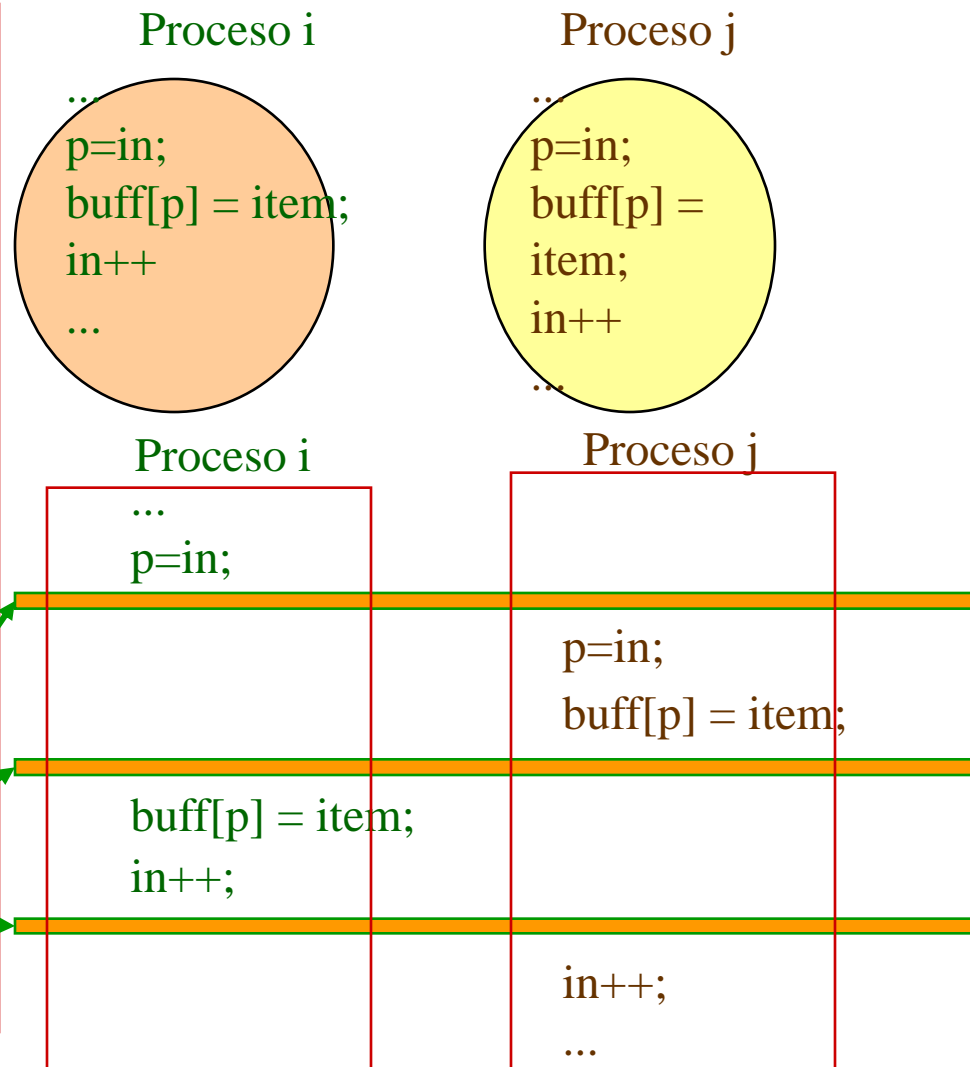
- ❑ Recursos Compartidos

- ❑ Mecanismos para Sincronización y Comunicación entre procesos.

7.2. Acceso exclusivo y secciones críticas.

- Comunicación:
 - Intercambio de información entre procesos.
- Sincronización:
 - Implícita (en algún mecanismo de comunicación)
 - Explícita. Funciones específicas de sincronización.
- Problema de la Sección Crítica (SC)
 - Condiciones de carrera

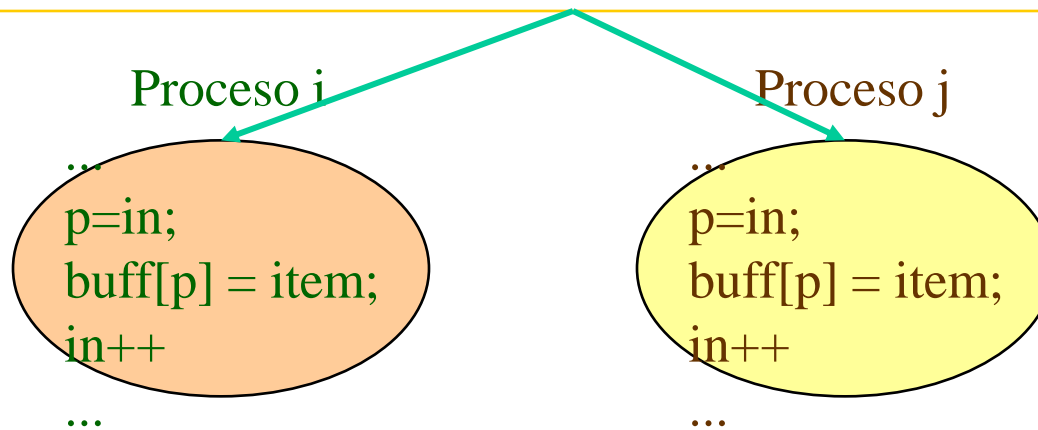
Cambio de contexto



Acceso exclusivo y secciones críticas.

Sección Crítica (SC):

Parte de código en el que se accede a un/unos recurso/s compartido/s entre varios procesos.



- Problema: **asegurar acceso exclusivo** a la misma sección crítica (SC)
- Cuando un proceso está ejecutando en una SC, ningún otro proceso puede ejecutarse en la misma SC, es decir, asegurar que las instrucciones de una SC, parezcan una única instrucción.

Acceso exclusivo y secciones críticas.

Acceso exclusivo a las secciones críticas:

Sean $\{P_0, P_1, \dots, P_{n-1}\}$ n procesos que comparten un recurso R . Sea SC_r el código de cada proceso en el que se accede a dicho recurso R .

Cuando un P_i está ejecutando dentro de la sección crítica SC_r , ningún otro proceso P_j puede entrar en su SC_r hasta que P_i abandone la suya.

Protocolo de acceso a una SC:

...

Entrar_CSr() /* Si nadie está dentro seguir, sino, esperar */

...

Salir_SCr()

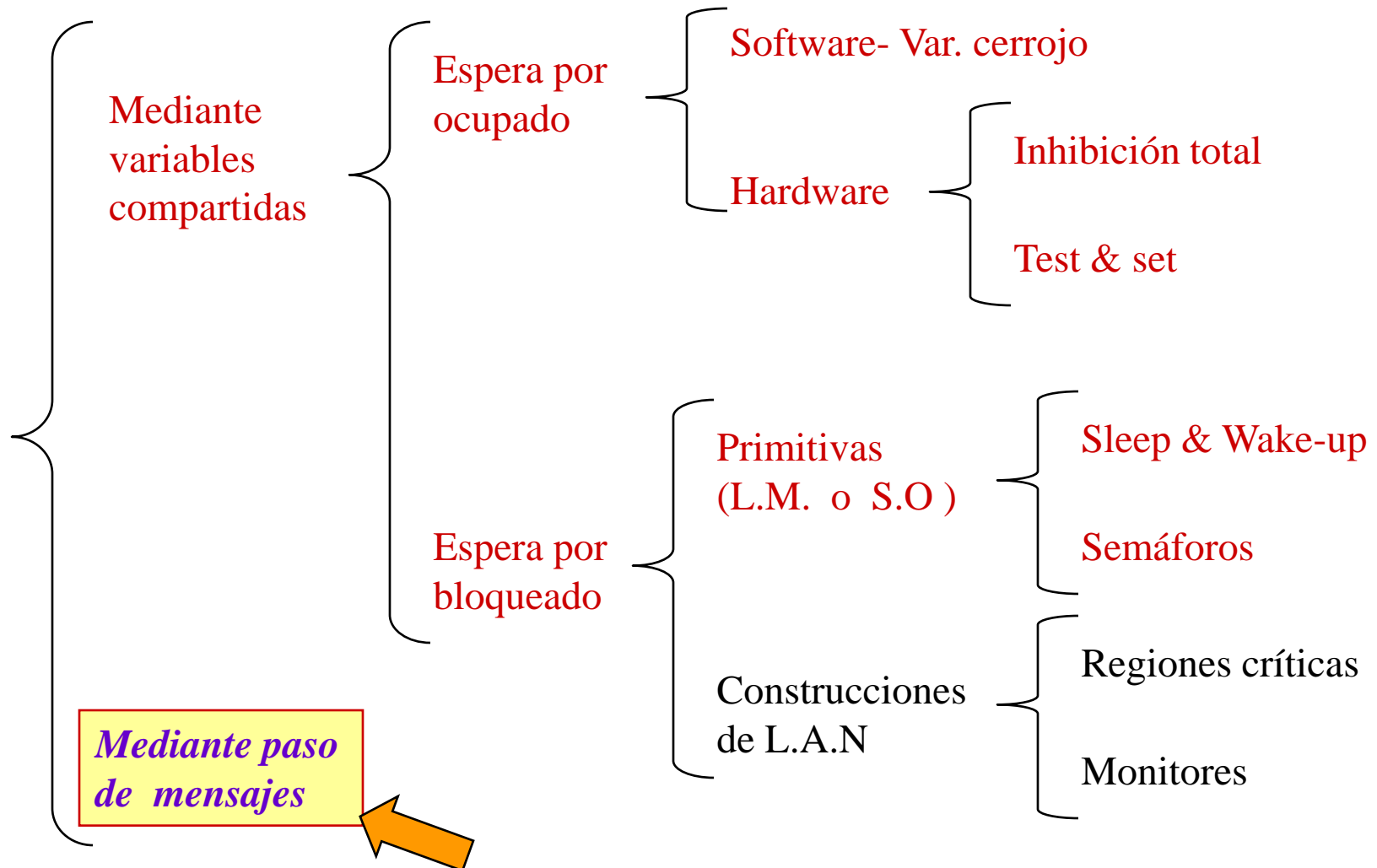
...

Acceso exclusivo y secciones críticas.

Condiciones para el acceso exclusivo a una SC:

- 1) **Exclusión mutua**
- 2) **No interbloqueo**
Ningún proceso detenido fuera de la SC puede impedir que otro entre en la misma.
- 3) **No inanición**
Un proceso no puede esperar indefinidamente para entrar en la SC
- 4) **Independencia del Hw**
No puede haber suposiciones sobre el número de los procesos o la velocidad de los mismos.

7.3 Mecanismos de Comunicación y Sincronización



7.4 Comunicación Mediante paso de mensajes

- Ofrece comunicación con sincronización (implícita)
- Sirve tanto para sistemas monoprocesador como multiprocesador y distribuidos.
- Se basa en dos primitivas o funciones
 - Send (destino, mensaje)
 - Receive(origen, mensaje)

El receptor no puede recibir un mensaje hasta que sea enviado por otro proceso.

- *Envío bloqueante ,recepción bloqueante.* Rendez-vous o entrega en mano, se bloquea tanto el emisor como el receptor hasta que se entrega el mensaje,
- *Envío NO bloqueante ,recepción bloqueante.* El receptor se bloquea hasta la llegada del mensaje. El emisor puede continuar.
- *Envío NO bloqueante ,recepción NO bloqueante.*
- Direccionamiento →

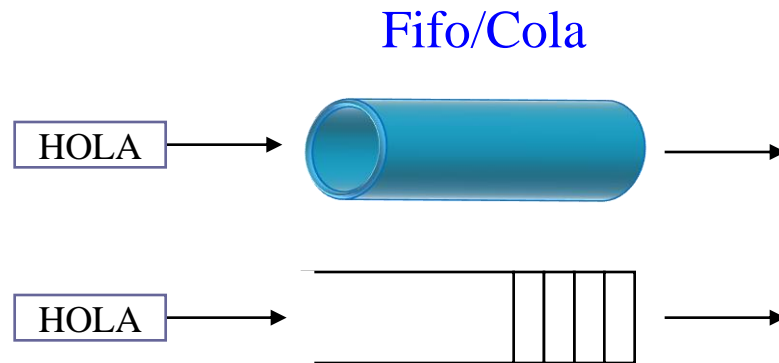
Comunicación Mediante paso de mensajes (Cont)

□ Direcccionamiento

- Directo. Identificador del destino o del origen.
 - Proceso X
 - Todos. Para enviar un mensaje a todos los preceso (broadcast)
 - Cualquiera (any). Para recibir un mensaje de cualquier proceso

- **Direcccionamiento indirecto.**

- A través de **buzones** (colas) o puertos.



7.5 Llamadas al sistema relacionadas con la comunicación entre procesos.

Buzones (colas FIFO)

Buzones con nombre:

```
#include <sys/types.h>
```

```
#include <sys/stat.h>
```

```
int mknod (char *path, int modo, int disp)
```

```
mknod( str, S_IFIFO | 0666, 0)
```

```
int mkfifo (char *path, int modo)
```

```
mkfifo(str, 0666)
```

Buzones (colas FIFO)(cont.)

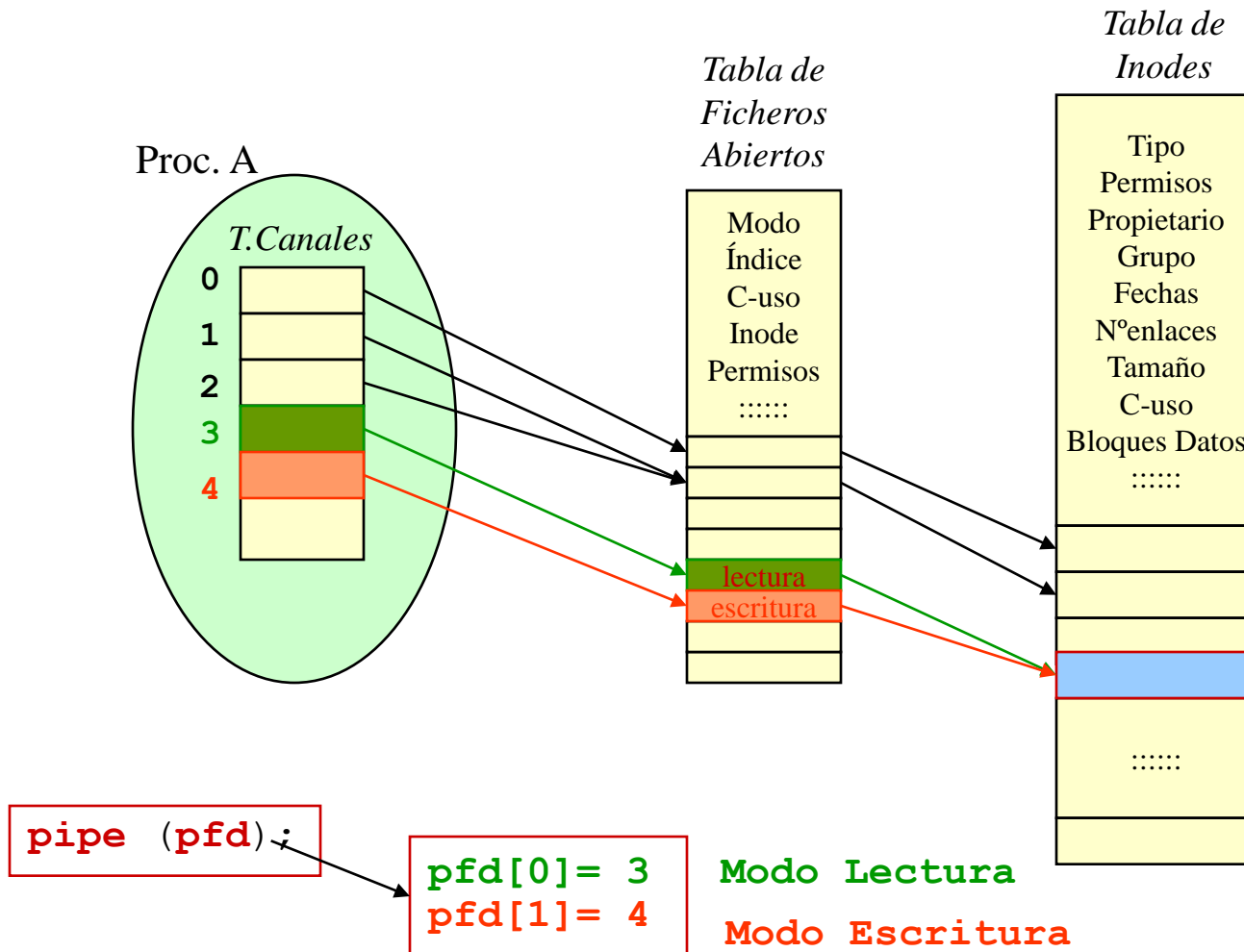
Buzones sin nombre (Pipes):

```
int pfd[2];  
  
int pipe (int *pfd)  
  
    int pfd[2];  
  
    pipe(pfd) ;  
    ...
```

Otras

```
int dup (int fd)  
  
int dup2 (int old_fd, int new_fd)
```

Pipe (cont.)



Dup (cont.)

