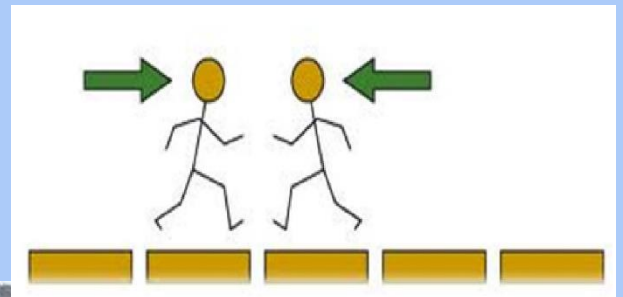
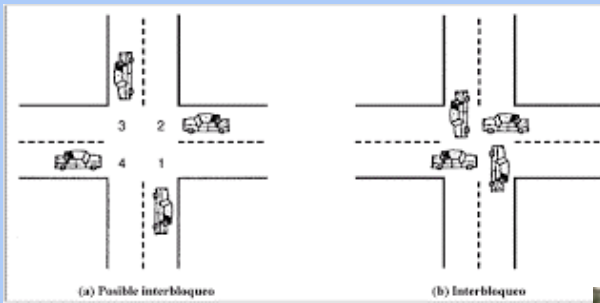


Unidad N° 7



BLOQUEO MUTUO (INTERBLOQUEO)



Contenidos



- 7.1. Modelo del sistema. 7.2. Caracterización de bloqueos mutuos. 7.3. Métodos para manejar bloqueos mutuos. 7.4. Recuperación. 7.5. Estrategias.

Introducción



Interbloqueo



- Deadlock
- Abrazo Mortal
- Bloqueo Mutuo

Un conjunto de procesos se encuentra en ***estado de interbloqueo*** cuando cada uno de ellos espera un suceso que solo puede originar otro proceso del mismo conjunto.

Recursos



- Un sistema se compone de un número finito de recursos que se distribuyen entre varios procesos que compiten por ellos.
- **Recursos**
 - Físicos
 - Lógicos
 - Expropiables
 - No Expropiables
- **Modo de operación**
 - Solicitud
 - Utilización
 - Liberación

Condiciones para que se produzca interbloqueo



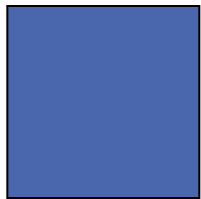
- Puede surgir si y sólo si en un sistema se presentan simultáneamente las condiciones:
 - Exclusión mutua.
 - Retención y espera
 - No expropiación
 - Espera circular
- No son completamente independientes

Modelado del interbloqueo

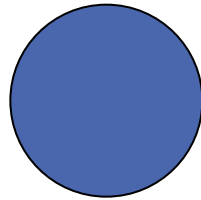


- Los interbloqueos pueden describirse utilizando un grafo dirigido y bipartido $G(N,A)$ llamado **grafo de asignación de recursos** que consta de 2 conjuntos: 1 de N nodos (vértices) y otro de A arcos.
- 2 tipos de nodos
 - Procesos
 - Recursos
- 2 tipos de arcos
 - **Arco de solicitud.** Es un arco que parte de un proceso P_i hacia un tipo de recurso R_j y se representa por $P_i \rightarrow R_j$ (o (P_i, R_j)). Significa que el proceso P_i solicitó una instancia del recurso R_j y se encuentra esperándolo.
 - **Arco de asignación.** Es un arco que sale de un tipo de recurso R_j y se dirige a un proceso P_i (representado por $R_j \rightarrow P_i$ o (R_j, P_i)). Significa que se ha asignado un ejemplar del tipo de recurso R_j al proceso P_i .

Modelado de interbloqueo



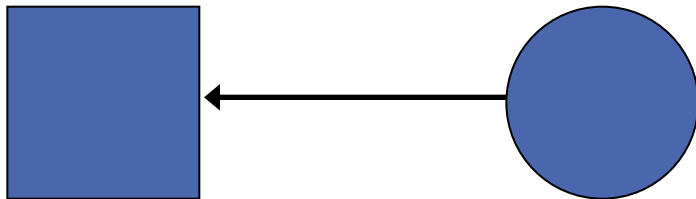
Recursos



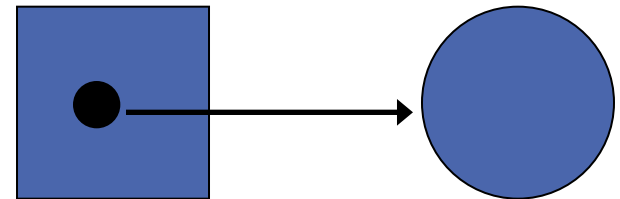
Procesos



Ejemplar de un Recurso

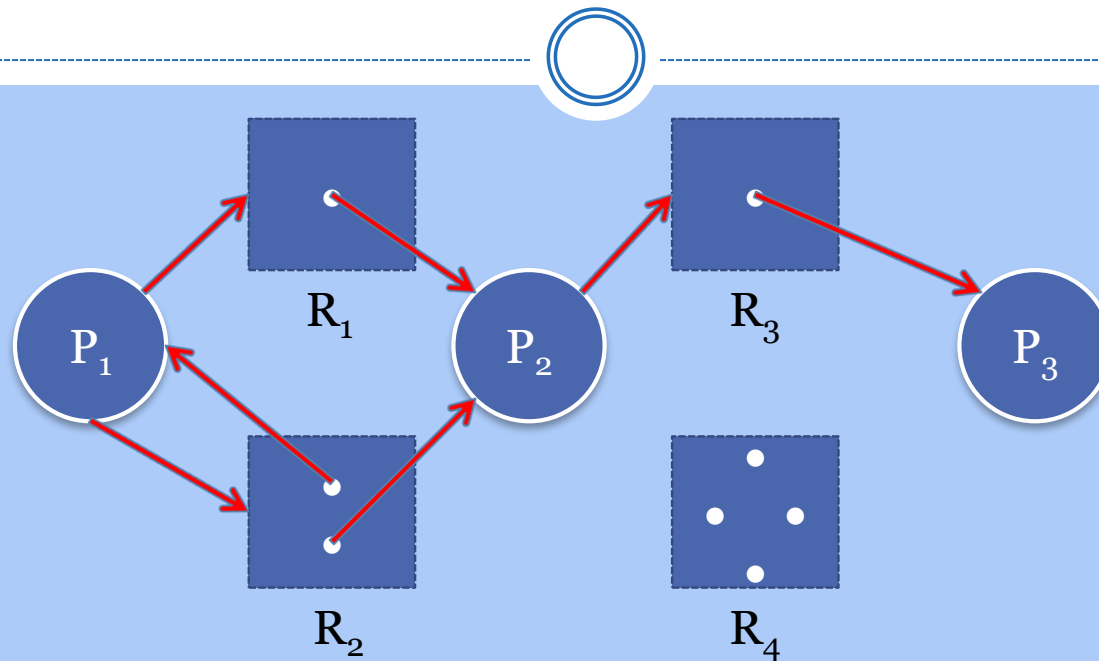


Arista de Solicitud de Recurso



Arista de Asignación de Recurso

Modelado del Interbloqueo



Estados de los procesos:

- El proceso P₁ tiene asignado un recurso de tipo R₂, y espera un recurso de tipo R₁ y un recurso de tipo R₂.
- El proceso P₂ tiene asignado un recurso de tipo R₁ y otro de tipo R₂, y espera un recurso de tipo R₃.
- El proceso P₃ tiene asignado un recurso de tipo R₃.

Modelado de Interbloqueo



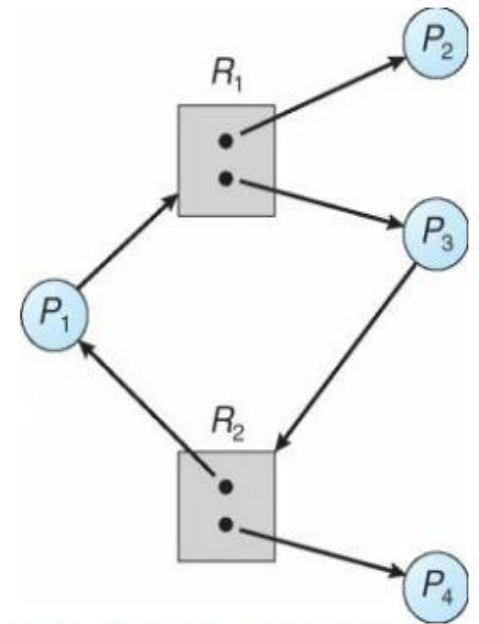
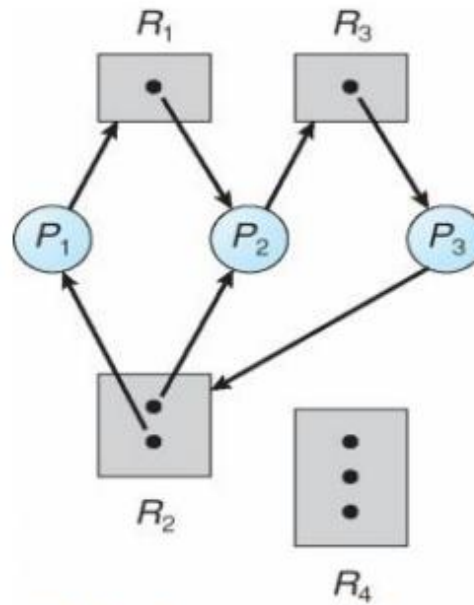
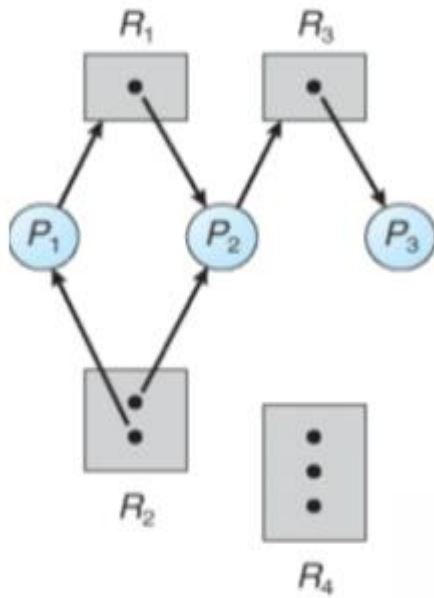
- Si el grafo de asignación de recursos **no contiene ciclos**, entonces ningún proceso del sistema se encuentra en interbloqueo.
- Si existe un ciclo, puede haber interbloqueo.

La presencia de un ciclo es condición necesaria pero no suficiente para la existencia de interbloqueo.

- Si de cada tipo de recurso existe un único ejemplar, entonces la presencia de un ciclo determina que existe interbloqueo.

En este caso la existencia de un ciclo si es condición necesaria y suficiente para la existencia de interbloqueo.

Modelado del Interbloqueo



Métodos para el tratamiento del Interbloqueo



ESTRATEGIAS

IGNORAR el problema

→ **Algoritmo del Avestruz**

↓
AFRONTAR el problema

Garantizar que nunca se entra en un estado interbloqueo

→ ✓ **Prevención**

✓ **Evitación**

Permitir que el sistema entre en interbloqueo y luego se recupere

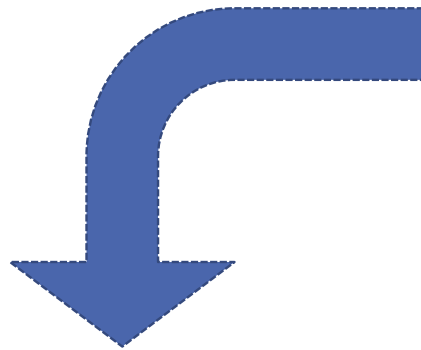
→ ✓ **Detección**

+

✓ **Recuperación**

Métodos para el tratamiento del Interbloqueo

PREVENCION



Objetivo: conseguir que sea imposible la aparición de situaciones de interbloqueo

Impedir que se produzca una de las cuatro condiciones necesarias para producirlo

- **Exclusión mutua**
- **Retención y espera**
- **No expropiación**
- **Espera circular**

Métodos para el tratamiento del Interbloqueo

PREVENCIÓN



EXCLUSIÓN MUTUA

- Existen recursos para los que no es posible negar la condición de exclusión mutua
 - ✦ **Archivos:** permiten múltiples accesos de lectura, pero únicamente un proceso puede escribir a la vez
- Es posible eliminar la condición de exclusión mutua en algunos recursos.
 - ✦ **Impresora:** Spooler

Métodos para el tratamiento del Interbloqueo

PREVENCIÓN



RETENCIÓN Y ESPERA

- Forzar a un proceso a que libere todos los recursos que retiene cada vez que solicite un recurso que no esté disponible.

Los procesos que esperan no tienen recursos y los que tienen no esperan.

- **Posibilidad 1:**
 - El proceso solicita todos los recursos necesarios antes de comenzar su ejecución.
 - ✦ Pobre utilización de recursos
 - ✦ Reducción del nivel de concurrencia

Métodos para el tratamiento del Interbloqueo

PREVENCIÓN



RETENCIÓN Y ESPERA

- Posibilidad 2

El proceso solicita los recursos de forma incremental a lo largo de su ejecución, pero libera todos los recursos retenidos si se encuentra con una negativa.

→ ✓ Menos Exigente

→ ✗ Los cambios hechos sobre la memoria o sobre archivos pueden corromper el sistema si no se concluyen

- Inconveniente adicional de **ambos** métodos

→ ✗ Pueden conducir a la postergación indefinida (o **inanición**) de algunos procesos que solicitan recursos muy utilizados

Métodos para el tratamiento del Interbloqueo

PREVENCIÓN



NO EXPROPIACIÓN

- Permitir que el sistema revoque la propiedad de ciertos recursos a los procesos bloqueados.
- La expropiación es involuntaria
 - ✦ El sistema operativo debe encargarse de salvar el estado y restaurarlo
 - Aún más difícil de llevar a cabo que la liberación voluntaria
- Este esquema se puede aplicar sobre recursos cuyo estado puede ser salvado y restaurado posteriormente (registros de CPU, memoria).

Métodos para el tratamiento del Interbloqueo

PREVENCIÓN



ESPERA CIRCULAR

- Ordenación lineal de los diferentes tipos de recursos del sistema.
- Los recursos del sistema se dividen en diferentes clases C_j ,
- Los interbloqueos se previenen con 2 restricciones:
 - exigiendo que todos los procesos soliciten y adquieran sus recursos en un orden estrictamente creciente de las clases mencionadas.
 - La adquisición de todos los recursos pertenecientes a una misma clase debe efectuarse con una sola petición (no incrementalmente)

Métodos para el tratamiento del Interbloqueo

PREVENCIÓN

ESPERA CIRCULAR

Pro

- No se pueden producir ciclos.
- El respeto de los procesos a la ordenación prescrita puede ser comprobada en tiempo de compilación.

Contras

- No se aprovechan adecuadamente los recursos.
 - Se adquieren en el orden prescrito y no se solicitan cuando realmente se necesitan

Métodos para el tratamiento del Interbloqueo

EVITACION

IDEA BÁSICA



Conceder únicamente las peticiones de recursos disponibles que no conduzcan a estados propensos al interbloqueo.

Petición



**ASIGNADOR
DE
RECURSOS**

estado seguro

CONCEDE

estado potencialmente inseguro

SUSPENDE

Examinar dinámicamente el estado de asignación de recursos para asegurar que no pueda presentarse la condición de espera circular

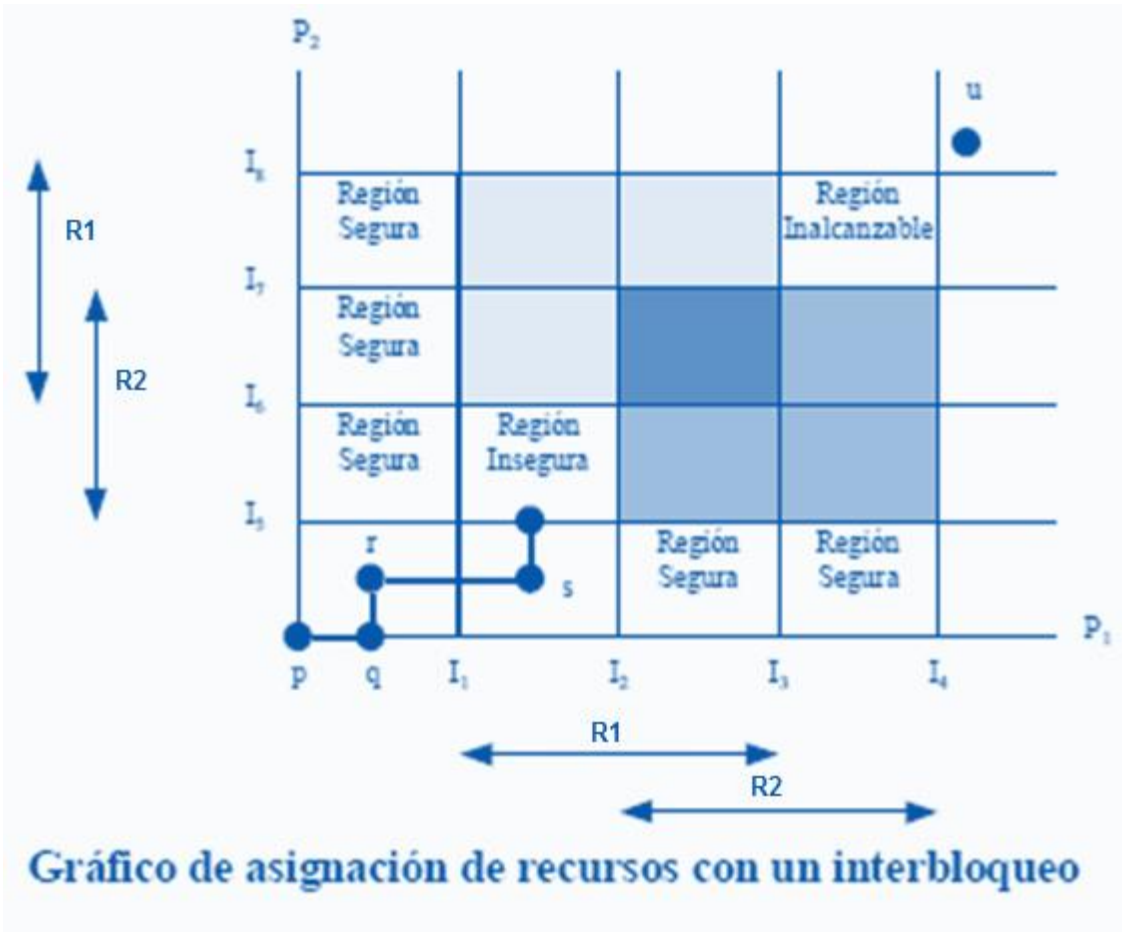
**ESTADO
SEGURO**

Métodos para el tratamiento del Interbloqueo

EVITACION



Trayectoria de Recursos



Métodos para el tratamiento del Interbloqueo

EVITACION



Un **estado** es **seguro** si el sistema puede asignar recursos a cada proceso hasta alcanzar el máximo de sus necesidades siguiendo algún orden arbitrario y aún así evitar el interbloqueo.

Un sistema se encuentra en estado **seguro** sólo si existe una secuencia segura.

Una secuencia de ejecución de procesos $\langle P_1, P_2, \dots, P_n \rangle$ es **segura** para el estado actual de asignación si para cada proceso P_i , los recursos que aún puede solicitar P_i pueden satisfacerse con los recursos actualmente disponibles más los retenidos por todos los P_j , tales que $j < i$.

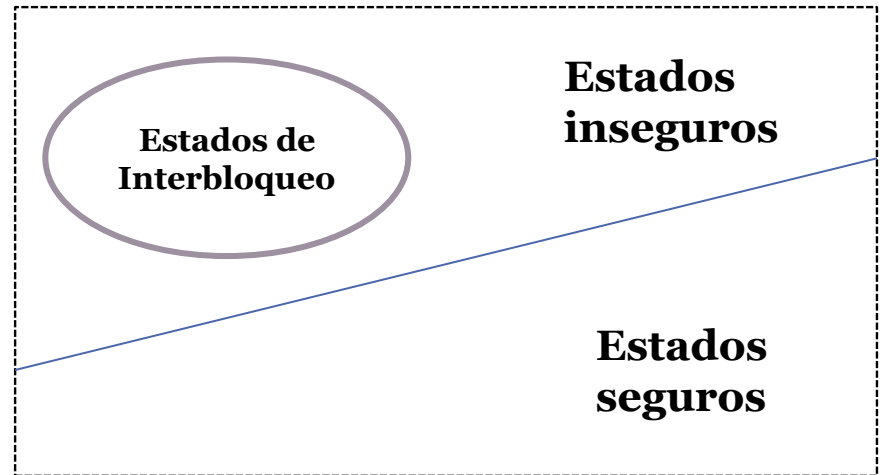
Si no existe esta secuencia, se dice que el estado es **inseguro**.

Métodos para el tratamiento del Interbloqueo

EVITACION



Estado inseguro
 \neq
Estado de interbloqueo



El algoritmo de evitación más famoso se debe a **Dijkstra** y se conoce con el nombre de **algoritmo del banquero**.

Métodos para el tratamiento del Interbloqueo

EVITACION



Algoritmo del Banquero

- Cuando un proceso nuevo ingresa en el sistema, debe declarar el ***número máximo de ejemplares*** de cada tipo de recursos que podría necesitar.
- Cuando un usuario solicita un conjunto de recursos, el sistema debe determinar si la asignación de esos recursos dejaría el sistema en un **estado seguro**. Si es así, los recursos se asignan; si no, el proceso deberá esperar hasta que otro libere los recursos suficientes.

Algoritmo del Banquero

EVITACION



- Sean n el número de procesos y m , el número de tipos de recursos. Se requieren las siguientes estructuras:
 - **Disponible:** Vector de orden m . Muestra la cantidad de ejemplares libres de cada tipo de recurso.
 - **Max:** Matriz de orden $n \times m$. Contiene la demanda máxima de cada proceso por cada tipo de recurso
 - **Asignación:** Matriz de orden $n \times m$. Contiene la cantidad de ejemplares de cada tipo de recurso que tiene asignado el proceso
 - **Necesidad:** Matriz de orden $n \times m$. Muestra la cantidad de ejemplares de cada tipo de recurso que aún requiere para completar su ejecución.

Algoritmo del Banquero

EVITACION



- **Algoritmo de Seguridad:** permite determinar si un sistema está o no en un estado seguro.
- 1. Sean Trabajo y Fin, vectores con longitud m y n. Asignar los valores iniciales Trabajo:=Disponible y Fin[i]:=false para $i=1,2,\dots,n$.
- 2. Buscar una i tal que
 - Fin[i] = false, y Necesidad_i ≤ Trabajo
 - Si no existe tal i, continuar con el paso 4
- 3. Trabajo := Trabajo + Asignación_i
Fin[i] = true
Ir al paso 2
- 4. Si **Fin[i] = true** para toda i, el sistema está en un **Estado Seguro**.

Algoritmo del Banquero

EVITACION



- **Algoritmo de Solicitud de Recursos:** Sea $Solicitud_i$, el vector de solicitudes del proceso P_i . Si $Solicitud_i[j] = k$, el proceso P_i quiere k ejemplares del tipo de recursos R_j . Cuando P_i solicita recursos, se emprenden las siguientes acciones:
 1. Si $Solicitud_i \leq Necesidad_i$, ir al paso 2. En caso contrario, indicar una condición de error (el proceso ha excedido su reserva máxima)
 2. Si $Solicitud_i \leq Disponible$, ir al paso 3. En caso contrario, P_i deberá esperar, ya que los recursos no están disponibles.
 3. Hacer que el sistema simule haber asignado al proceso P_i los recursos que solicitó modificando el estado como sigue:
 - $Disponible := Disponible - Solicitud_i$;
 - $Asignación_i := Asignación_i + Solicitud_i$;
 - $Necesidad_i := Necesidad_i - Solicitud_i$;Si el estado resultante es seguro, la transacción se llevará a cabo, si es inseguro, el sistema deberá volver a su antiguo estado.

Algoritmo del Banquero

EVITACION



Sea un sistema con cinco procesos, P1 a P5, y tres tipos de recursos, A, B y C, de los que existen 10, 5 y 7 ejemplares, respectivamente. Supongamos que en el instante actual tenemos la siguiente situación del sistema:

	Asignación			Max			Disponible			Necesidad		
	A	B	C	A	B	C	A	B	C	A	B	C
P ₁	0	1	0	7	5	3	3	3	2	7	4	3
P ₂	2	0	0	3	2	2				1	2	2
P ₃	3	0	2	9	0	2				6	0	0
P ₄	2	1	1	2	2	2				0	1	1
P ₅	0	0	2	4	3	3				4	3	1

- Determinar si este estado es seguro o inseguro.
- Supongamos ahora que P2 hace una petición $Solicitud_2 = (1, 0, 2)$. ¿Cómo actuaría el algoritmo del banquero para decidir si esta petición puede atenderse?
- Determinar si las siguientes solicitudes pueden atenderse o no:
 - Solicitud de (3, 3, 1) por parte de P5.
 - Solicitud de (0, 2, 0) por parte de P1.

Métodos para el tratamiento del Interbloqueo

EVITACION



PROBLEMA

- Sean tres procesos P1, P2 y P3 y un único tipo de recurso del que existe un total de 10 unidades. En el estado inicial P1 tiene 3 instancias del recurso, pero podría necesitar en algún instante hasta 9; P2 tiene 2 pero podría necesitar 4 más adelante y P3 tiene 2 pero podría necesitar 7. En ese instante quedan libres 3 unidades.
 - Demostrar que el estado inicial es seguro.
 - Demostrar que si P1 obtiene un ejemplar del recurso el estado resultante es inseguro.

Métodos para el tratamiento del Interbloqueo

EVITACION



RESUMEN

- No requiere la adquisición de todos los recursos de una vez.
- No aparece el problema de la infrautilización de recursos debido a la adquisición prematura
- Requiere que se haga la pre-reclamación de las necesidades máximas de recursos.

Métodos para el tratamiento del Interbloqueo

EVITACION



RESUMEN

- No necesita la expropiación de recursos.
- Impone costos de almacenamiento y computación para detectar los estados seguros.
- El algoritmo del banquero, aunque en teoría funciona perfectamente, en la práctica no es útil.
 - *Los procesos rara vez conocen de antemano sus necesidades máximas de recursos.*
 - *El número de procesos no es fijo, como precisa el algoritmo (por ejemplo, cuando se conectan y desconectan los usuarios).*

Métodos para el tratamiento del Interbloqueo

DETECCION



IDEA BÁSICA

1. Conceder libremente los recursos disponibles a los procesos solicitantes



2. Examinar el sistema para determinar si se ha producido un interbloqueo



3. Intentar romperlo

Métodos para el tratamiento del interbloqueo

DETECCION



Método Práctico:

1. Intentar reducir el grafo de asignación de recursos suprimiendo todas las retenciones y peticiones de todo proceso cuyas solicitudes puedan ser concedidas.
2. ¿El grafo queda completamente reducido?
 1. **NO: El sistema está interbloqueado y los procesos que quedan son los afectados**
 2. **SI: el sistema no está interbloqueado**

La capacidad que tiene el algoritmo para identificar los procesos interbloqueados es también importante, pues facilita la recuperación posterior.

Métodos para el tratamiento del Interbloqueo

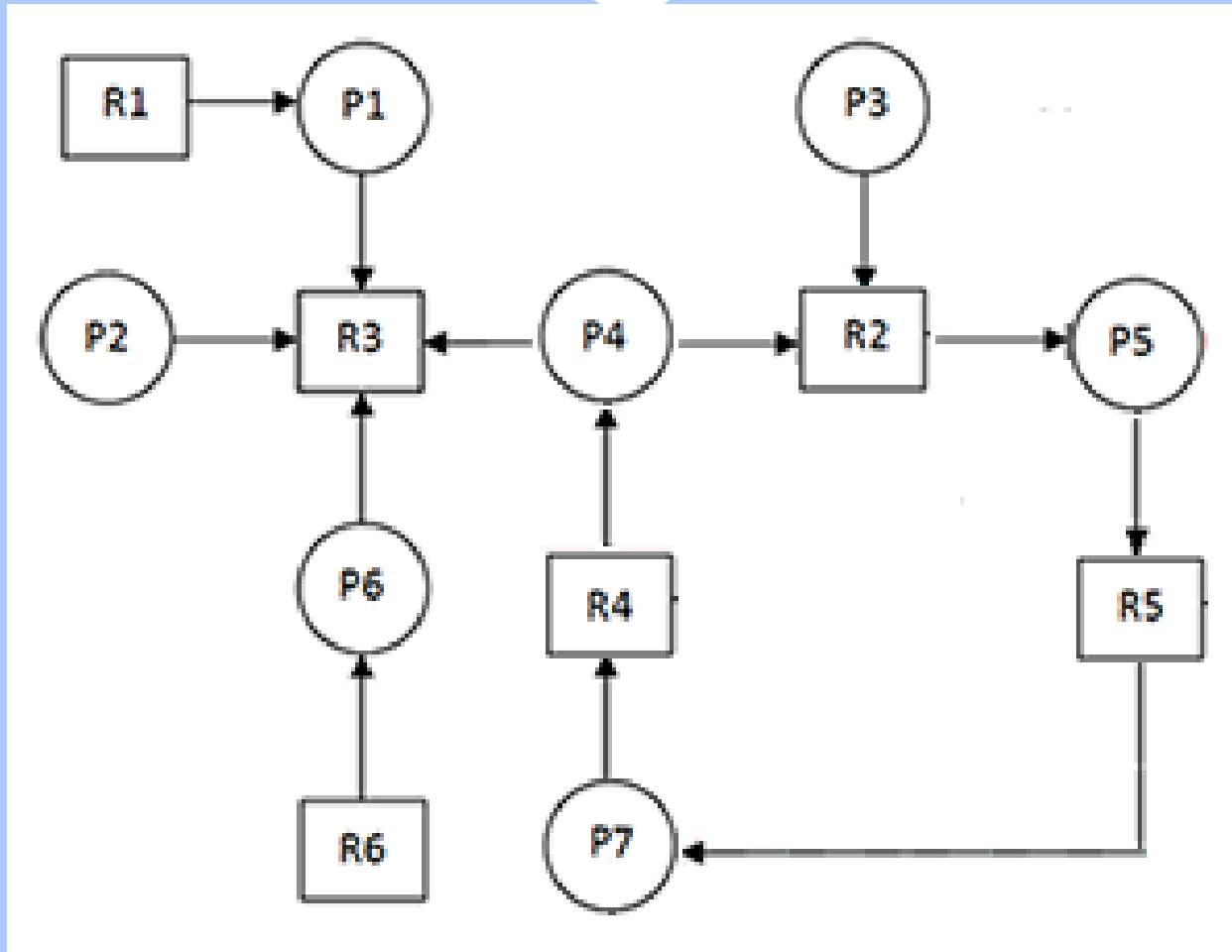
DETECCION



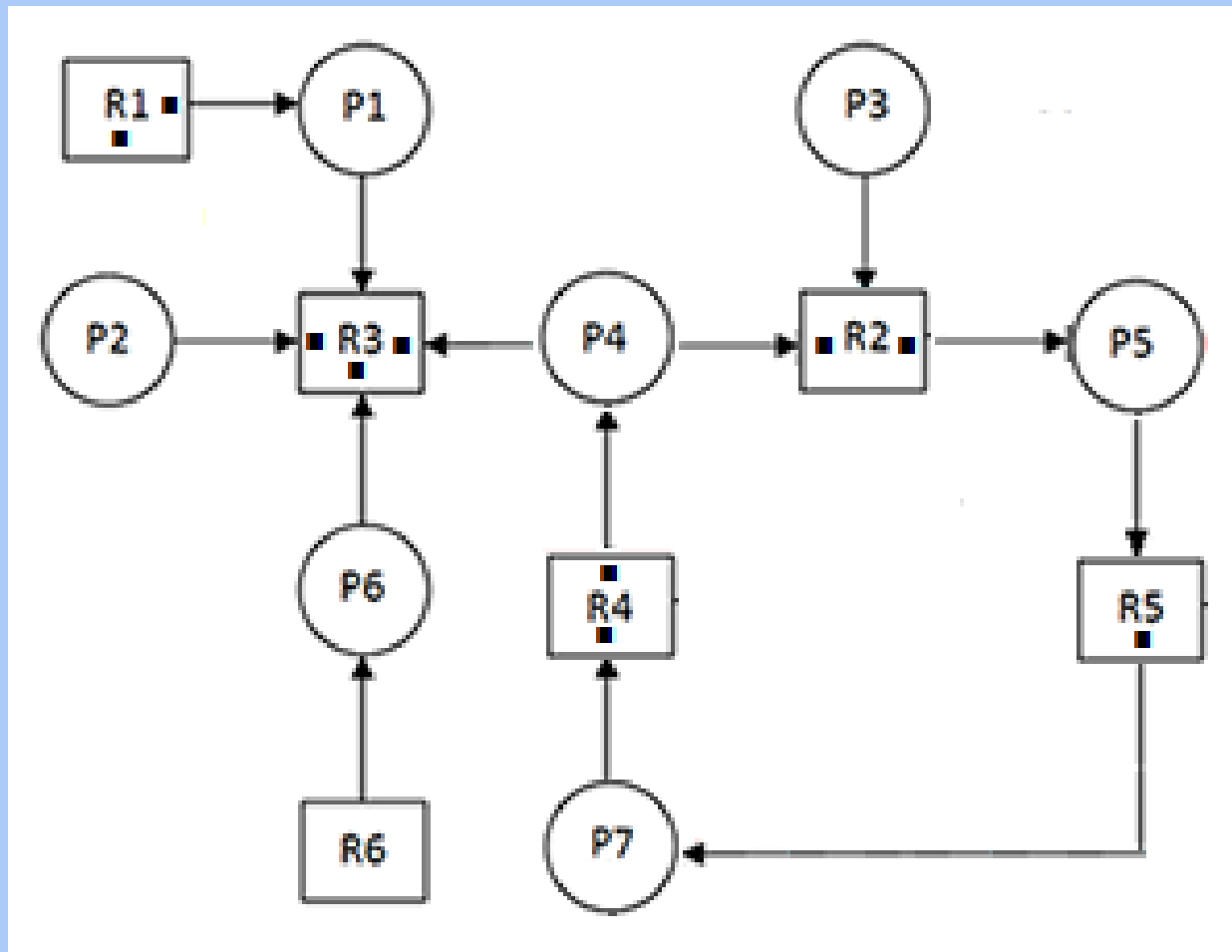
Reducción de Gráfica de Asignación de Recursos.

- Si las peticiones de recursos de un proceso pueden ser concedidas, se dice que una gráfica puede ser reducida por ese proceso.
- La reducción de una gráfica por un proceso determinado se muestra retirando:
 - Las flechas que van de los recursos al proceso (los recursos asignados al proceso).
 - Las flechas que van del proceso al recurso (las peticiones actuales del proceso).
- Si una gráfica puede ser reducida por todos sus procesos, entonces no hay interbloqueo.
- Si una gráfica no puede ser reducida por todos sus procesos, entonces los procesos “irreducibles” constituyen la serie de procesos interbloqueados de la gráfica.

Reducción de Gráfica ¿Hay Bloqueo?



Reducción de Gráfica ¿Hay Bloqueo?



Métodos para el tratamiento del Interbloqueo

DETECCION



Algoritmo de Detección de Bloqueos de Forma “Varios Recursos de Cada Tipo”:

Se considera:

- Disponible: Un vector de longitud m . Nro de recursos disponibles en cada tipo.
- Asignación: Una matriz $n \times m$, define el nro de recursos de cada tipo que ya se han asignado a cada proceso.
- Solicitud: Una matriz $n \times m$ indica la solicitud actual de cada proceso.

Métodos para el tratamiento del Interbloqueo

DETECCION



- Este algoritmo investiga todas las posibles secuencias de asignación para los procesos que todavía no se llevan a cabo.
1. Sean **Trabajo** y **Fin** vectores con longitud m y n .
Trabajo := Disponible. Si Asignación _{i} $\neq 0$ entonces Fin[i]:=false; si no, Fin[i]:=true;
 2. Buscar un índice i tal que:
Fin[i] = False y Solicitud \leq Trabajo. Si no existe ir a 4.
 3. Trabajo := Trabajo + Asignación _{i} ; Fin[i] := true . Ir al paso 2
 4. Si Fin[i] = false; para alguna i , $1 \leq i \leq n$, el sistema está en estado de bloqueo mutuo. Si Fin[i] = False, el proceso P_i está en bloqueo mutuo.

Métodos para el tratamiento del Interbloqueo

DETECCION

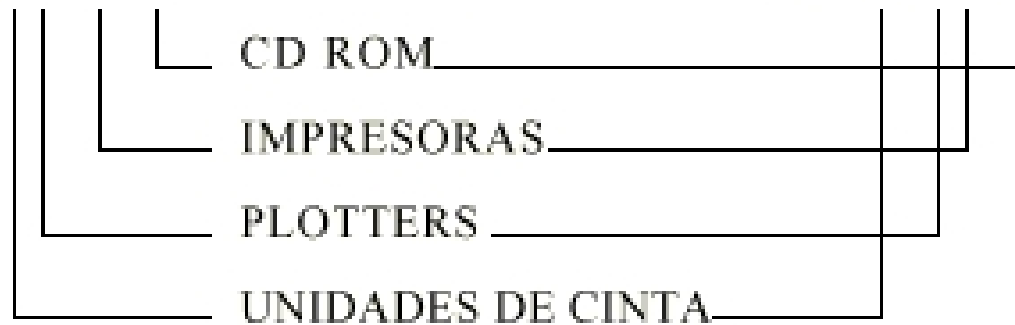


RECURSOS EN EXISTENCIA:

$$E = (4 \ 2 \ 3 \ 1)$$

RECURSOS DISPONIBLES:

$$A = (2 \ 1 \ 0 \ 0)$$



MATRIZ DE ASIGNACION ACTUAL:

$$C = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 2 & 0 & 0 & 1 \\ 0 & 1 & 2 & 0 \end{bmatrix}$$

MATRIZ DE SOLICITUDES:

$$R = \begin{bmatrix} 2 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 2 & 1 & 0 & 0 \end{bmatrix}$$

Métodos para el tratamiento del Interbloqueo

DETECCION



- Uso del algoritmo

Quando invocar al algoritmo?

- Depende de:
 - La frecuencia con la que ocurren los bloqueos.
 - La cantidad de procesos a los que afectará el bloqueo si ocurre.

Métodos para el tratamiento del Interbloqueo

RECUPERACION



- Cuando un algoritmo de detección determina que existe un interbloqueo, existen varias alternativas para tratar de eliminarlo:
 - Informar al operador del sistema
 - ✦ Resolución manual
 - El sistema rompe el interbloqueo y se recupera automáticamente
 - ✦ Abortar uno o más procesos para romper la espera circular
 - ✦ Expropiar algunos recursos de uno o más de los procesos implicados

Métodos para el tratamiento del Interbloqueo

RECUPERACION



Terminación De Procesos

- Abortar todos los procesos interbloqueados
 - Simple
 - Costo muy elevado → Trabajo Realizado por los procesos
- Abortar procesos de uno en uno hasta eliminar el ciclo
 - Después de abortar cada proceso, hay que invocar al algoritmo de detección para saber si aún se mantiene el interbloqueo
 - ↓
 - Mucho tiempo de proceso adicional

Métodos para el tratamiento del Interbloqueo

RECUPERACION



Terminación De Procesos

- En cualquiera de los casos, puede que no sea fácil terminar un proceso (p.e. si se encuentra actualizando un archivo).
- Cuando se utiliza el método incremental, se presenta un nuevo problema de política o decisión: selección de la víctima

Métodos para el tratamiento del Interbloqueo

RECUPERACION



Terminación De Procesos

- Desde un punto de vista económico, se debe seleccionar aquel proceso cuya terminación represente el costo mínimo.
- Existen muchos factores que pueden determinar el proceso a eliminar, incluyendo:
 - La prioridad del proceso.
 - Tiempo de ejecución ya consumido y tiempo de ejecución previsto.
 - Cuántos recursos y de qué tipo está usando el proceso.
 - Cuántos recursos más necesita el proceso para concluir.
 - Cuántos procesos es preciso terminar.
 - Tipo del proceso (por lotes o interactivo).

Métodos para el tratamiento del Interbloqueo

RECUPERACION



Expropiación de Recursos

- Quitar sucesivamente los recursos de los procesos que los tienen y asignarlos a otros que los solicitan hasta conseguir romper el interbloqueo.
- Hay que considerar tres aspectos:
 - Selección de la víctima
 - Retroceso (rollback)
 - Postergación indefinida

Métodos para el tratamiento del Interbloqueo

RECUPERACION



RESUMEN

- La detección y recuperación de interbloqueos proporciona un mayor grado potencial de concurrencia que las técnicas de prevención o de evitación.
- La sobrecarga en tiempo de ejecución de la detección puede ser un parámetro ajustable en el sistema.
- La recuperación de interbloqueos puede ser atractiva en sistemas con una baja probabilidad de interbloqueos.
- En sistemas con elevada carga, la concesión sin restricciones de peticiones de recursos puede conducir a frecuentes interbloqueos

Resumen: Prevención, Predicción (Evitación) y Detección



Principio	Política de Reparto de Recursos	Esquemas	Ventajas Principales	Desventajas Principales
Prevención	Conservadora; los recursos están poco ocupados.	Solicitud de todos los recursos a la vez	<ul style="list-style-type: none"> • Funciona bien con procesos que realizan una sola ráfaga de actividad. • No es necesaria la apropiación. 	<ul style="list-style-type: none"> • Ineficiencia. • Retrasos en el inicio de los procesos.
		Apropiación	<ul style="list-style-type: none"> • Conveniente cuando se aplica a recursos cuyo estado se puede salvar y restaurar fácilmente. 	<ul style="list-style-type: none"> • Apropia más habitualmente de lo necesario. • Sujeto a reinicios cíclicos.
		Ordenación de recursos	<ul style="list-style-type: none"> • Aplicación factible por medio de chequeos durante la compilación. • No hace falta procesamiento durante la ejecución, pues el problema se resuelve durante el diseño del sistema. 	<ul style="list-style-type: none"> • No permite las solicitudes incrementales de recursos. • Poco uso de la apropiación.
Detección	Muy liberal; los recursos solicitados se conceden cuando es posible.	Comprobación periódica del interbloqueo.	<ul style="list-style-type: none"> • Nunca retrasa el inicio de un proceso • Facilita el manejo en línea 	<ul style="list-style-type: none"> • Pérdidas inherentes a la apropiación.
Predicción	A medio camino entre la detección y la prevención.	Manipulación para encontrar al menos un camino seguro.	<ul style="list-style-type: none"> • No es necesaria la apropiación. 	<ul style="list-style-type: none"> • Deben conocerse las demandas futuras de recursos. • Los procesos pueden bloquearse durante largos periodos.

Métodos para el tratamiento del Interbloqueo



Estrategia Combinada Para El Manejo De Interbloqueos

- Ninguno de los métodos presentados es adecuado para ser utilizado como estrategia exclusiva de manejo de interbloqueos en un sistema complejo.
- Dividir los recursos del sistema en una colección de clases disjuntas y aplicando el método más adecuado de manejo de interbloqueo a los recursos de cada clase particular.
- La ordenación de recursos puede servir para prevenir interbloqueos entre clases

