



Universidad Nacional del Nordeste



Facultad de Ciencias Exactas y Naturales y Agrimensura

Cátedra: Sistemas Operativos

Año:2024

Trabajos Prácticos 2da. Parte

Alumno:

DNI:

- |                                       |          |
|---------------------------------------|----------|
| • Benites, Matias Maximiliano         | 42098665 |
| • Gómez, Sebastián Exequiel           | 42603108 |
| • Más González, Fernando              | 37157720 |
| • Orban, Tobías Naim                  | 46385637 |
| • Ristovich, Mauro Ezequiel Alejandro | 41380447 |

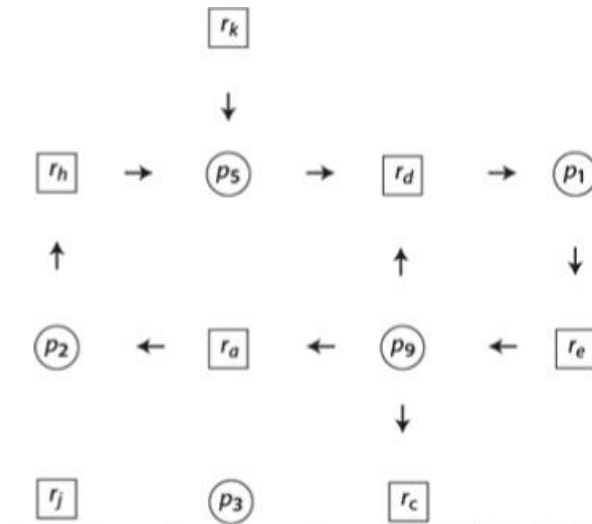
## Trabajo Práctico 5 - Bloqueos

1)

a- Si se produce bloqueo ya que visualizamos en el gráfico un ciclo

$L: (rh, p5, rd, p1, re, p9, ra, p2, rh)$ .

b- Si es posible hacer una reducción de la gráfica ya que  $p3$  se puede ejecutar con  $rj$ , es decir, ya está libre.

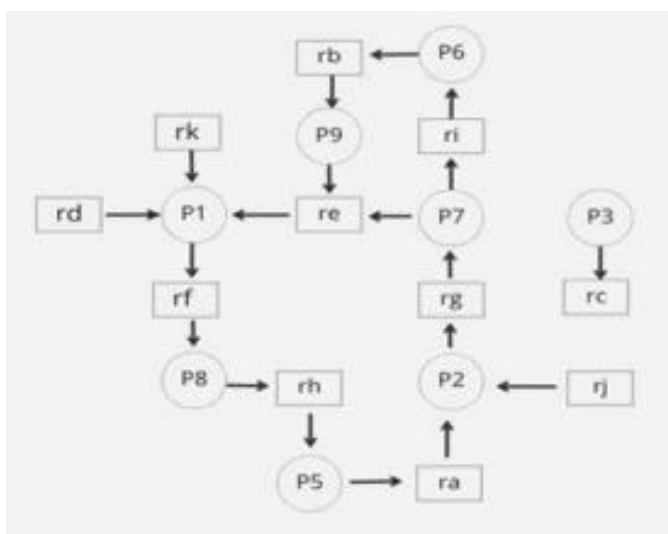


c- Los procesos bloqueados son:  $(p5, p1, p9, p2)$ .

d- El bloqueo podría solucionarse mediante la apropiación del recurso  $rd$ .

e-  $L\{p5, p2, p9, p1\}$

2- a

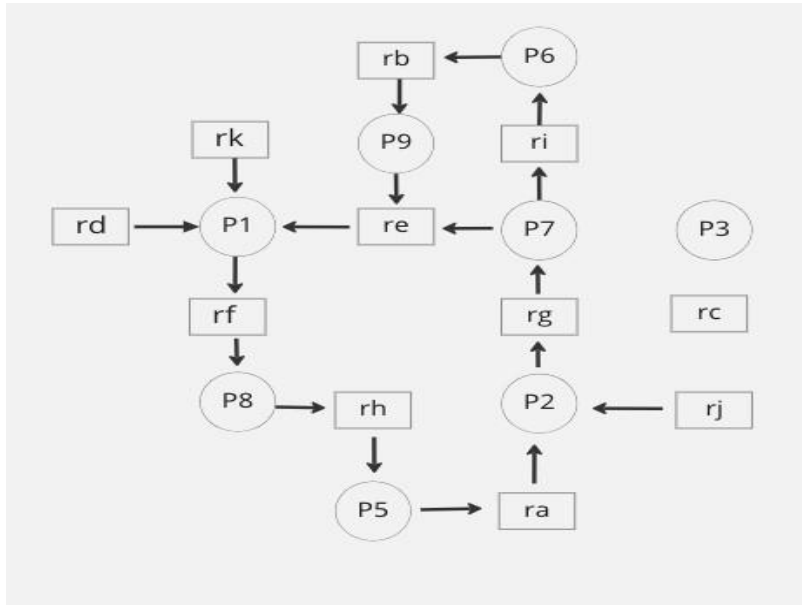


b- Si se produce bloqueo ya que visualizamos en el gráfico dos ciclos

1.  $L: (p1, rf, p8, rh, p5, ra, p2, rg, p7, rl, p6, rb, p9, re, p1).$

2.  $L: (p1, rf, p8, rh, p5, ra, p2, rg, p7, re, p1).$

c- Si es posible hacer una reducción de la gráfica ya que p3 se puede ejecutar con rc.

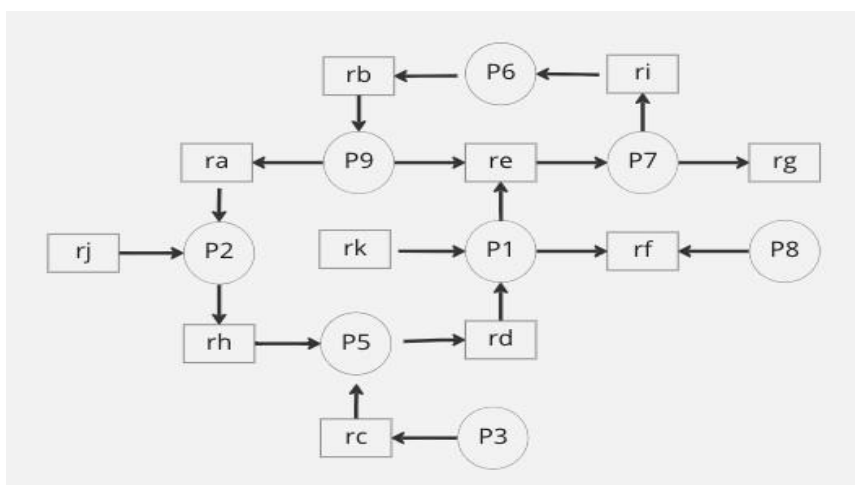


d- Los procesos bloqueados son:  $(p1, p8, p5, p2, p7, p6, p9)$

e- El bloqueo podría solucionarse mediante la apropiación del recurso re.

f-  $L\{p9, p6, p7, p2, p5, p8,$

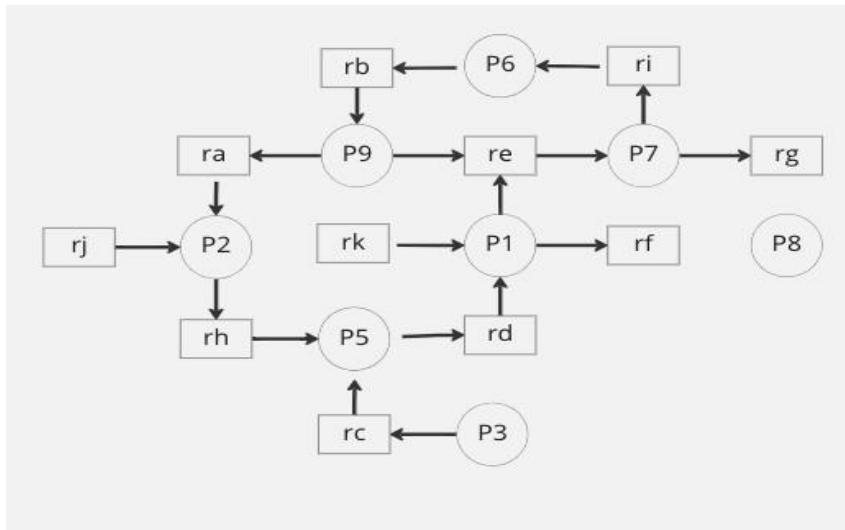
$p1\}$  3- a



b- Si se produce bloqueo ya que visualizamos en el gráfico dos ciclos L:(p7, ri, p6, rb, p9, re, p7).

L:(p1, re, p7, ri, p6, rb, p9, ra, p2, rh, p5, rd, p)

c--Si es posible hacer una reducción de la gráfica ya que p8 se puede ejecutar con rf.



d- Los procesos bloqueados son: (p7, p6 , p9, p2, p5, p1).

e- El bloqueo podría solucionarse mediante la apropiación del recurso **re**.

f- A {p1, p5, p3} B:{p1, p5, p2, p9, p6, p7}

4-

Existencia 5 6 3 4 7					
Disponibles:	3	1	2	2	4

Matriz de Asignación 1 1 0 1 0								Matriz de solicitudes: 2 1 2 2 4				
	0	2	0	0	1			3	4	5	3	5
	0	1	0	0	1			3	2	2	2	3
	1	1	1	1	1			2	2	1	1	5

El proceso 1 puede satisfacer la solicitud de recursos por lo que se ejecuta y los libera.

Disponible 3 1 2 2 4					
Solicitud	2	1	2	2	4

Se actualizan los recursos disponibles.

3 1 2 2 4					
	1	1	0	1	0
Disponible	4	2	2	3	4

Para el proceso 2, no se puede satisfacer ya que no dispone de 4 unidades de impresoras.

Disponible	4	2	2	3	4
Solicitud	3	4	5	3	5

El proceso 3 puede satisfacer la solicitud de recursos por lo que se ejecuta y los libera.

Disponible	4	2	2	3	4
Solicitud	3	2	2	2	3

Se actualizan los recursos disponibles.

	4	2	2	3	4
	0	1	0	0	1
Disponible	4	3	2	3	5

El proceso 4 puede satisfacer la solicitud de recursos por lo que se ejecuta y los libera. Se actualizan los recursos disponibles:

Disponible	4	3	2	3	5
Solicitud	2	2	1	1	5

Se actualizan los recursos disponibles:

	4	3	2	3	5
	1	1	1	1	1
Disponible	5	4	3	4	6

El proceso 2 sigue sin poder satisfacer la solicitud de recursos. Por ende el proceso está bloqueado.

Disponible	5	4	3	4	6
Solicitud	3	4	5	3	5

Si cambiamos la matriz de solicitudes:

3 1 2 2 2				
3	2	2	1	3
4	4	3	4	1

2 2 1 1 5
-----------

El proceso 1 puede satisfacer la solicitud de recursos por lo que se ejecuta y los libera.

Disponible 3 1 2 2 4					
Solicitud	3	1	2	2	2

Se actualizan los recursos disponibles.

3 1 2 2 4					
	1	1	0	1	0
Disponible	4	2	2	3	4

El proceso 2 puede satisfacer la solicitud de recursos por lo que se ejecuta y los libera.

Disponible	4	2	2	3	4
Solicitud	3	2	2	1	3

Se actualizan los recursos disponibles.

4 2 2 3 4					
	0	2	0	0	1
Disponible	4	4	2	3	5

El proceso 3, no se puede satisfacer ya que no dispone de 3 unidades de cintas.

Disponible	4	4	2	3	5
Solicitud	4	4	3	4	1

El proceso 4 puede satisfacer la solicitud de recursos por lo que se ejecuta y los libera.

Disponible	4	4	2	3	5
Solicitud	2	2	1	1	5

Se actualizan los recursos disponibles.

	4	4	2	3	5
	1	1	1	1	1
Disponible	5	5	3	4	6

El proceso 4 puede satisfacer la solicitud de recursos por lo que se ejecuta y los libera.

Disponible 5 5 3 4 6					
Solicitud	4	4	3	4	1

Se actualizan los recursos disponibles.

5 5 3 4 6					
	0	1	0	0	1
Disponible	5	6	3	4	7

El vector de recursos disponibles es igual al vector de recursos existentes. El algoritmo finaliza y no se han detectado bloqueos.

5-

Existencia 7 4 5 7 6 3						
Disponibles:	3	2	4	1	3	1

Matriz de Asignación 1 1 0 2 0 0							Matriz de solicitudes: 3 3 4 3 1 1						
	2	0	0	1	1	0		3	4	5	3	5	2
	0	1	0	2	1	1		3	2	4	2	3	1
	1	0	1	1	1	0		2	2	4	1	2	1

El proceso 1, no se puede satisfacer ya que no dispone de 3 unidades de Plotters.

Disponible 3 2 4 1 3 1						
Solicitud	3	3	4	3	1	1

El proceso 2, no se puede satisfacer ya que no dispone de 4 unidades de Plotters.

Disponible 3 2 4 1 3 1						
Solicitud	3	4	5	3	5	2

El proceso 3 no se puede satisfacer ya que no dispone de 2 impresoras.

Disponible 3 2 4 1 3 1						
Solicitud	3	2	4	2	3	1

El proceso 4 puede satisfacer la solicitud de recursos por lo que se ejecuta y los libera.

Disponible 3 2 4 1 3 1						
Solicitud	2	2	4	1	2	1



Se actualizan los recursos disponibles.

3 2 4 1 3 1						
	1	0	1	1	1	0
Disponible	4	2	5	2	4	1

El proceso 3 puede satisfacer la solicitud de recursos por lo que se ejecuta y los libera.

Disponible 4 2 5 2 4 1						
Solicitud	3	2	4	2	3	1

Se actualizan los recursos disponibles.

4 2 5 2 4 1						
	0	1	0	2	1	1
Disponible	4	3	5	4	5	2

El proceso 1 puede satisfacer la solicitud de recursos por lo que se ejecuta y los libera.

Disponible	4	3	5	4	5	2
Solicitud	3	3	4	3	1	1

Se actualizan los recursos disponibles.

	4	3	5	4	5	2
	1	1	0	2	0	0
Disponible	5	4	5	6	5	2

El proceso 2 puede satisfacer la solicitud de recursos por lo que se ejecuta y los libera.

Disponible 5 4 5 6 5 2						
Solicitud	3	4	5	3	5	2

Se actualizan los recursos disponibles.

5 4 5 6 5 2						
	2	0	0	1	1	0
Disponible	7	4	5	7	6	2

Como podemos observar, se han ejecutado todos los procesos de manera satisfactoria, pero al igualar el vector de disponibilidad y el de existencia podemos comprobar que no son iguales.

Si cambiamos la matriz de solicitudes:

3 2 2 1 3 1					
2	4	5	3	5	2
1	1	2	2	4	1
4	2	4	3	1	1

El proceso 1, no se puede satisfacer ya que no dispone de 3 unidades de Zip drivers.

Disponible 1 1 3 2 2 2						
Solicitud	3	2	2	1	3	1

El proceso 2, no se puede satisfacer ya que no dispone de 2 unidades de Zip drivers.

Disponible 1 1 3 2 2 2						
Solicitud	2	4	5	3	5	2

El proceso 3, no se puede satisfacer ya que no dispone de 4 impresoras.

Disponible 1 1 3 2 2 2						
Solicitud	1	1	2	2	4	1

El proceso 4, no se puede satisfacer ya que no dispone de 4 unidades de Zip drivers.

Disponible 1 1 3 2 2 2					
------------------------	--	--	--	--	--

Solicitud	4	2	4	3	1	1
-----------	---	---	---	---	---	---

Como podemos observar, no se pudo satisfacer ninguno de los procesos ya que no se tenían todos los recursos necesarios para hacerlo. Por ende todos los procesos están bloqueados.

6-

Existencia 4 2 5 6 2 3						
Disponibles:	1	1	3	2	2	2

Matriz de Asignación 2 0 0 2 0 1							Matriz de solicitudes: 1 1 3 1 1 1						
	1	0	2	0	0	0		3	1	2	3	4	3
	0	0	0	1	0	0		4	1	2	3	3	2
	0	1	0	1	0	0		3	3	2	1	1	3

El proceso 1 puede satisfacer la solicitud de recursos por lo que se ejecuta y los libera.

Disponible 1 1 3 2 2 2						
Solicitud	1	1	3	1	1	1

Se actualizan los recursos disponibles.

1 1 3 2 2 2						
	2	0	0	2	0	1
Disponibles:	3	1	3	4	2	3

El proceso 2 no se puede satisfacer ya que no dispone de 4 impresoras.

Disponible 3 1 3 4 2 3						
Solicitud	3	1	2	3	4	3

El proceso 3 no se puede satisfacer ya que no dispone de 4 Zip drivers.

Disponible 3 1 3 4 2 3						
Solicitud	4	1	2	3	3	2

El proceso 4 no se puede satisfacer ya que no dispone de 3 escáneres.

Disponible 3 1 3 4 2 3						
Solicitud	3	3	2	1	1	3

El proceso está bloqueado ya que al finalizar el algoritmo observamos que no pudimos ejecutar todos los procesos.

Si cambiamos la matriz de solicitudes:

Matriz de solicitudes: 1 1 3 1 1 1						
	3	2	2	3	2	3
	4	1	2	3	3	2
	3	1	2	1	1	3

El proceso 1 puede satisfacer la solicitud de recursos por lo que se ejecuta y los libera.

Disponibles 1 1 3 2 2 2						
Solicitud	1	1	3	1	1	1

Se actualizan los recursos disponibles.

1 1 3 2 2 2						
-------------	--	--	--	--	--	--

2 0 0 2 0 1						
Disponibles	3	1	3	4	2	3

El proceso 2 no se puede satisfacer ya que no dispone de 2 escáneres.

Disponibles 3 1 3 4 2 3						
Solicitud	3	2	2	3	2	3

El proceso 3 no se puede satisfacer ya que no dispone de 4 Zip drivers

Disponible 3 1 3 4 2 3						
Solicitud	4	1	2	3	3	2

El proceso 4 puede satisfacer la solicitud de recursos por lo que se ejecuta y los libera.

Disponible 3 1 3 4 2 3						
Solicitud	3	1	2	1	1	3

Se actualizan los recursos disponibles.

3 1 3 4 2 3						
	0	1	0	1	0	0
Disponible	3	2	3	5	2	3

El proceso 2 puede satisfacer la solicitud de recursos por lo que se ejecuta y los libera.

Disponible 3 2 3 5 2 3						
Solicitud	3	2	2	3	2	3

Se actualizan los recursos disponibles.

3 2 3 5 2 3						
	1	0	2	0	0	0
Disponible	4	2	5	5	2	3

El proceso 3 puede satisfacer la solicitud de recursos por lo que se ejecuta y los libera.

Disponible 4 2 5 5 2 3						
Solicitud	4	1	2	3	3	2

Se actualizan los recursos disponibles.

4 2 5 5 2 3						
	0	0	0	1	0	0
Disponible	4	2	5	6	2	3

## Trabajo Practico 6 – Sincronización

1)

T1		T2		T3
0		0		0
4		6		8
8		12		16
12		18		24
16		24		32
20		30		40
24		41		48
28		47		56
32		53		64
54		59		72
58		65		80
62		71		88

2)

a.

A		B		C
0		0		0
3		4		5
6		8		10
9		12		15
12		16		20
15		20		25
18		24		30
21		28		35
24		32		40
27		36		45
30		40		50
33		44		55
36		48		60

b.

**Send(M1, 6 )**

**Send(M2, 5 )**

**Send(M3, 16 )**

**Send(M4, 44 )**

**Send(M5, 30 )**

**Send(M6, 24 )**

**Send(M7, 40 )**

c. Al terminar de enviar los mensajes A marca un tiempo de reloj de 36, b 48 y C 60.

d. Se necesita usar el algoritmo de lamport.

e.

A		B		C
0		0		0
3		4		5
6		8		10
9	m1	12		15
12		16		20
15		20	m3	25
18		24		30
31	m5	28		35
34		32		40
37	m6	36		45
40		40		50
43	m7	44		55
46	m4	48		60

f. Al terminar de enviar los mensajes A marca un tiempo de reloj de 46, b 48 y C 60.

3)

X		R		N
0		0		0
6		2		4
12		4		8
18		6		12
24		8		16
30		17		20
36		19		24
42		21		28
48		23		32
54		25		49
60		27		53
66		29		57

4)





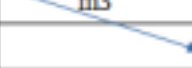

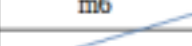



T4		T5		T6
0		0		0
7		1		8
14		2		16
21		3		24
28		4		32
35		5		40
42		6		48
49		43		56
56		44		64
63		45		72
70		46		80
77		47		88

5)

X1		F3		G5		C4
0		0		0		0
3		6		2		5
6		12		4		10
9		18		6		15
12		24		8		20
21		30		10		25
24		36		12		30
27		42		25		35
30		48		27		40
33		54		29		45
36		60		31		50
39		66		33		61



6)

T4		T0		T2
0		0		0
4	 m1	8		6
8		16		12
12		24		18
16	 m5	32		24
20	 m3	40		30
24		48		36
28		56	 m4	42
32	 m6	64		48
36		72	 m2	54
40		80		60

b.

Send(M1, 4 )

Send(M2, 54 )

Send(M3, 20 )

Send(M4, 52 )

Send(M5, 18 )

Send(M6, 60 )

c. Al terminar de enviar los mensajes A marca un tiempo de reloj de 40, b 80 y C 60.

d. Es necesario usar el algoritmo de lamport.

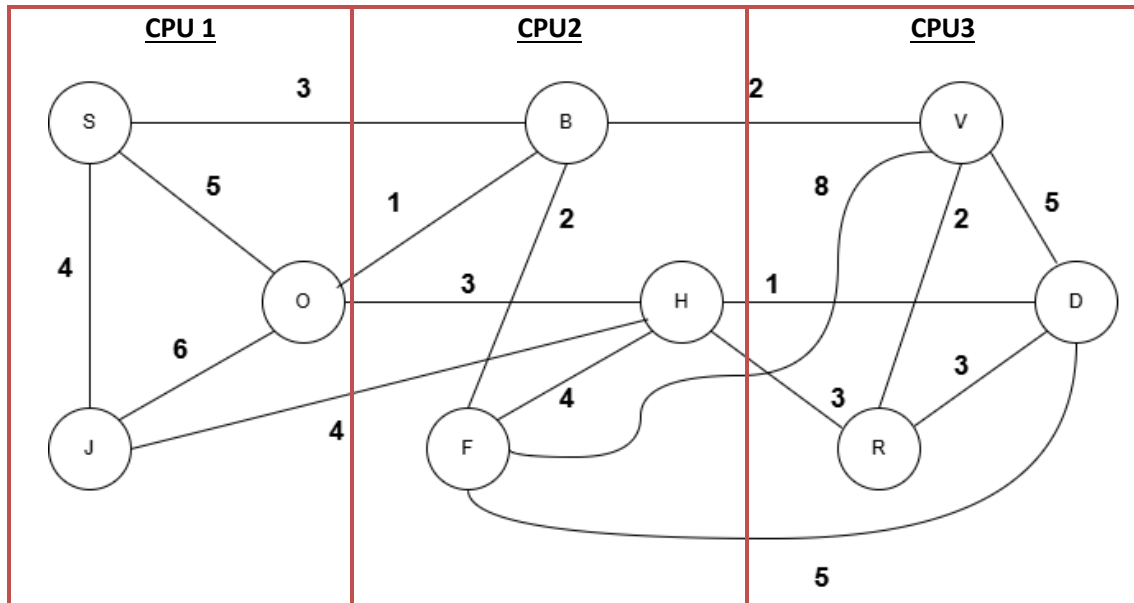
e.

T4		T0		T2
0		0		0
4	m1	8		6
8		16		12
12		24		18
19	m5	32		24
23	m3	40		30
27		48		36
31		56	m4	42
35	m6	64		57
65		72	m2	63
69		80		69

f. Al terminar de enviar los mensajes A marca un tiempo de reloj de 69, b 80 y C 69.

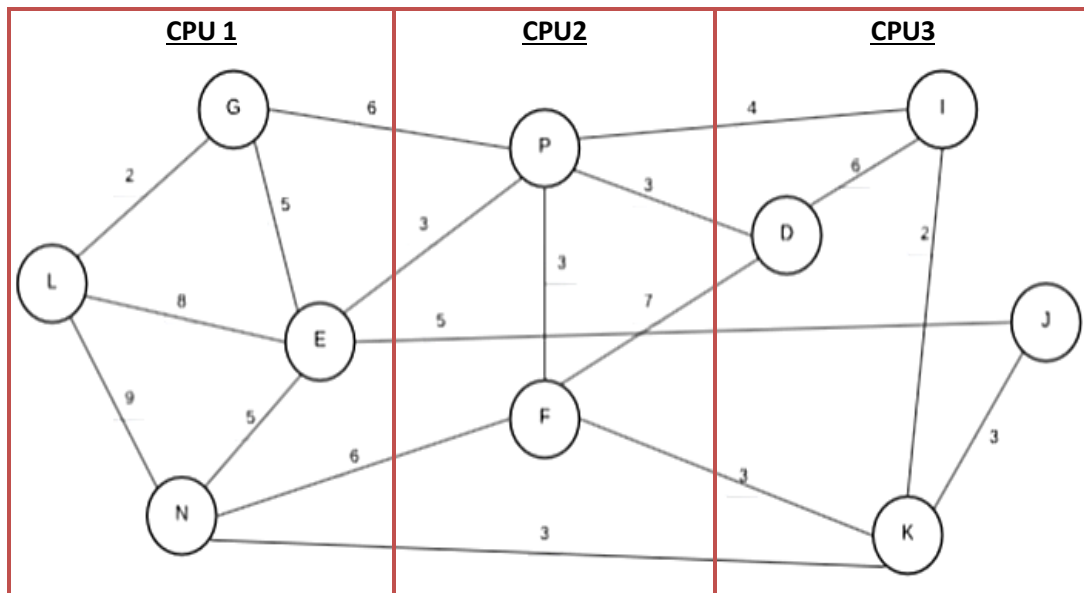
## TP 7 – Procesos y Procesadores en Sistemas Distribuidos

1)



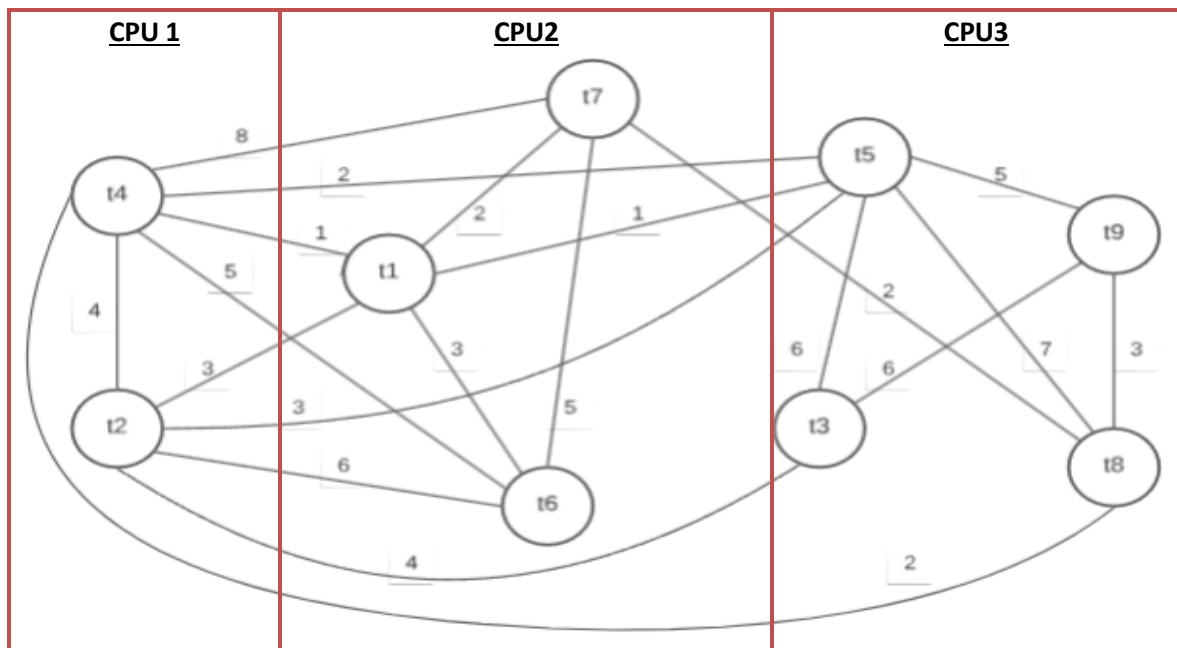
Total Tráfico de la red:  $3 + 4 + 1 + 3 + 8 + 1 + 3 + 2 + 5 = 30$

2)



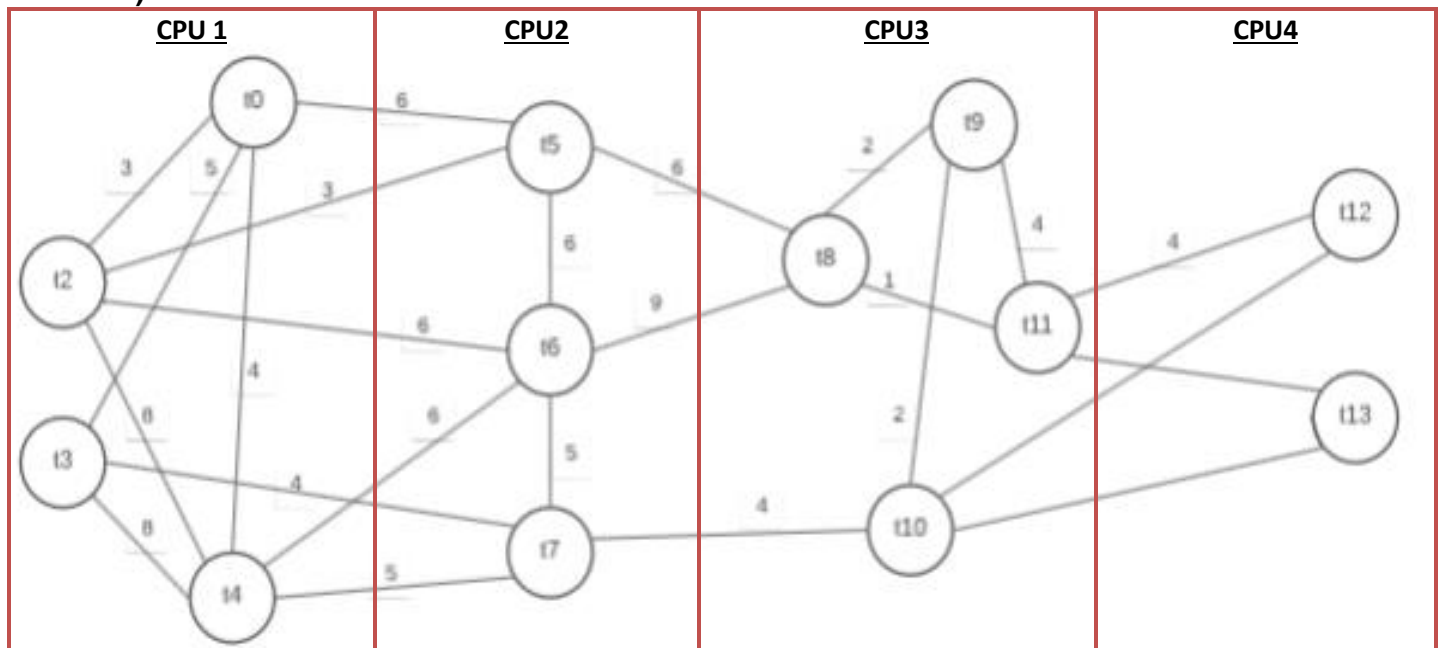
Total Tráfico de la red:  $6 + 3 + 5 + 6 + 3 + 3 + 7 + 3 + 4 = 40$

3)



Total, Trafico de la red:  $8 + 2 + 1 + 5 + 3 + 3 + 6 + 4 + 2 + 1 + 2 = 37$

4)



Tráfico total de la red:  $6 + 3 + 6 + 6 + 4 + 5 + 6 + 9 + 4 + 6 + 5 + 5 + 4 + 2 = 71$

### **Trabajo Práctico 8 – Sistemas Distribuidos de Archivos**

A continuación, se presenta, mediante un cuadro comparativo, las principales características relacionadas a los sistemas de archivos tradicionales y distribuidos:

<b>Aspectos</b>	<b>Sistema de archivos tradicionales</b>	<b>Sistema de archivos distribuidos</b>
<b>Diseño</b>	Se basan en un modelo centralizado, donde los datos se almacenan en un solo lugar y se acceden a través de una red. Más fáciles y simples de implementar. Más simples pero menos escalables.	Se basan en un modelo descentralizado, donde los datos se almacenan en múltiples lugares y se acceden a través de una red. Son más escalables y tolerante a fallos. Más complejos que los tradicionales. Los clientes deben verlo como un sistema de archivos normal (Convencional).
<b>Seguridad</b>	Poseen mayores riesgos de pérdida de datos por fallos del sistema. Tienen una menor complejidad en la seguridad por su diseño. Se basa en un control de acceso centralizado con medidas de seguridad más uniformes y coherentes con todo el sistema. Menor exposición a diversos riesgos de red. Implanta medidas de seguridad estándar como permisos de archivos y directorio.	Tiene una complejidad mayor de seguridad debido a su diseño de datos distribuidos. Dada la descentralización se puede dificultar la implementación de medidas de seguridad coherentes en todos los nodos del sistema. Expuesto a ataques de interceptación, manipulación o denegación de servicios. Se utiliza el cifrado y protocolos seguros de comunicación para proteger los datos durante la transferencia.
<b>Disponibilidad</b>	Posee un único punto de fallo, si el servidor central falla, todos los servicios asociados y los archivos no estarán disponibles. El tiempo de inactividad puede ser prolongado en caso de fallas, ya que la recuperación del sistema y restauración de datos lleva tiempo y produce una interrupción prolongada de los servicios. Su límite de escalabilidad también puede afectar la disponibilidad.	Alta disponibilidad debido a la redundancia y replicación de datos. Si un nodo falla, los demás están disponibles. Esto ayuda a garantizar la disponibilidad de datos incluso en presencia de fallos (tolerancia a fallos). Posee un balanceo de cargas de trabajo entre múltiples nodos y dispositivos de almacenamiento que mejora el rendimiento evitando cuellos de botella.

		incrementando la disponibilidad general del sistema. Permite una recuperación rápida y sin interrupciones mediante la activación de las réplicas o la redistribución automática de datos.
<b>Servicio de Archivos y Directorios</b>	Servicios estándar de archivos y directorios almacenados en un único sistema conectado directamente a la computadora. El ser de acceso local posee menor latencia.	Servicios distribuidos con acceso a través de múltiples servidores. Mayor latencia al poseer archivos distribuidos.
<b>Implantación</b>	El aspecto clave de la implantación del almacenamiento de archivos es el registro de los bloques asociados a cada archivo. Las estrategias de asignación más comunes son: Contigua y no contigua, de la cual derivan esquemas de almacenamientos más dinámicos (Asignación encadenada orientada hacia el sector, asignación por bloque, lista ligada, encadenamiento de bloques de índices, etc.)	Incluye aspectos tales como: El uso de los archivos, estructura del sistema, ocultamiento, duplicación (replica) y control de la concurrencia. Utiliza asignación de una lista enlazada con tablas en memoria. Existen varias copias de archivos y cada una está en un servidor de archivos independiente.
<b>Recuperación ante Fallos</b>	Se basa en una estrategia de respaldo donde uno pueda tener un BACKUP de los archivos que puedan ser recuperados en caso de fallas críticas del sistema como: errores humanos, eventos climatológicos, robos, golpes que afecten a la unidad de almacenamiento dejándola dañada y ataques que puedan provocar pérdida de datos. Se debe garantizar la disponibilidad de los datos por medio de cualquier mecanismo.	Para que sean tolerante a fallos se requiere cada vez más redundancia de hardware, comunicaciones, software, etc. La réplica de los archivos es un requisito esencial, aunque se contempla la posibilidad de que los sistemas funcionen aun con la carencia de una parte de los datos. Los tiempos de fallo aceptables por los usuarios son cada vez menores.

--	--	--

### **Ejemplos de tecnologías que se apliquen a entornos distribuidos en la actualidad:**

#### **1) Colossus (Google File System):**

Fuente: [A peek behind Colossus, Google's file system | Google Cloud Blog](#)

Este sistema de almacenamiento es Colossus, que respalda el amplio ecosistema de servicios de almacenamiento de Google, como Cloud Storage y Firestore, que admite una amplia gama de cargas de trabajo, incluido el procesamiento de transacciones, el servicio de datos, el análisis y el archivado, los discos de arranque y los directorios de inicio.

#### **2) AMAZON EMR**

Fuente: [Características de Amazon EMR - Plataforma para big data - Amazon Web Services](#)

Con Amazon EMR puede utilizar varios almacenes de datos, incluidos Amazon S3, Hadoop Distributed File System (HDFS) y Amazon DynamoDB. Amazon EMR ha realizado numerosas mejoras en Hadoop para que pueda procesar grandes volúmenes de datos almacenados en Amazon S3 sin problemas.

#### **3) HDFS**

Fuente: [¿Qué es HDFS? Sistema de archivos distribuido Apache Hadoop | IBM](#)

HDFS es un sistema de archivos distribuido que maneja grandes conjuntos de datos que se ejecutan en hardware básico. Se utiliza para escalar un solo clúster de Apache Hadoop a cientos (e incluso miles) de nodos.

### **Bibliografía Consultada:**

- [1] La Red Martinez, D. L. *Sistemas Operativos*. U.N.N.E – ARGENTINA
- [2] A. S. Tanenbaun. *Sistemas operativos modernos. 3ra Edición*. Pearson Educación, México, 2009
- [3] W. Stirling. *Sistemas Operativos. 5ta Edición*. Pearson Educación, S.A., Madrid, 2005.