

- Estructura de una partición
- Estructura de un nodo-i
- Entradas de directorio
- Tamaños estándar
- **Ejercicios**

• Ejercicio 1: Tamaño máximo de un archivo

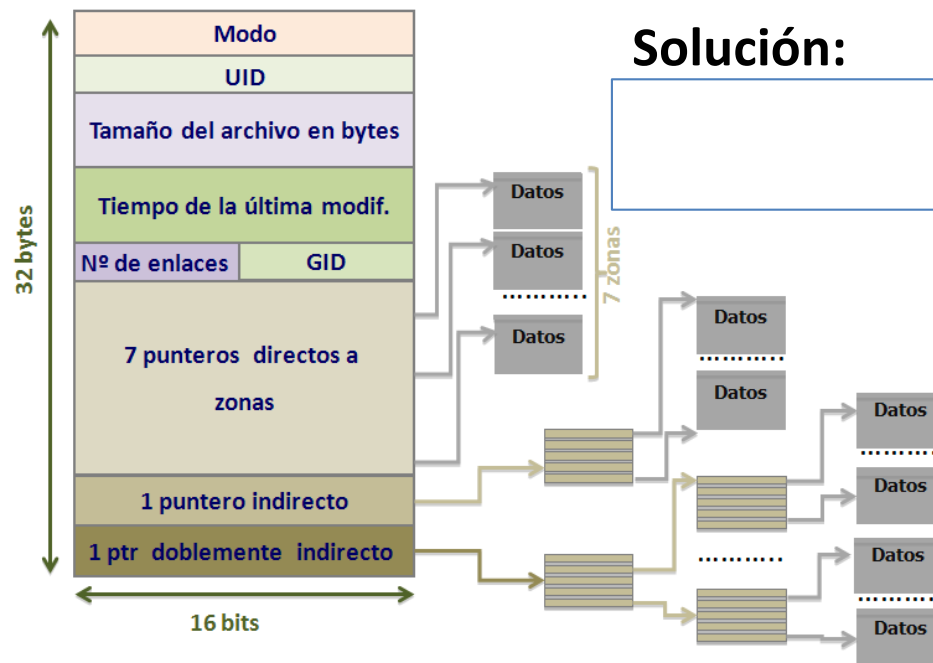
Determine el máximo tamaño teórico (no tenga en cuenta el tamaño real del dispositivo sobre el que deba residir el archivo) de un archivo Minix. Especifique los bloques direccionados con cada tipo de puntero. Los parámetros de Minix ha considerar son:

Punteros a zonas de datos de 16 bits

Tamaño de bloque 1K

1 zona = 1 bloque

Estructura de nodo-i: 7 punteros directos, 1 indirecto y 1 doble indirecto



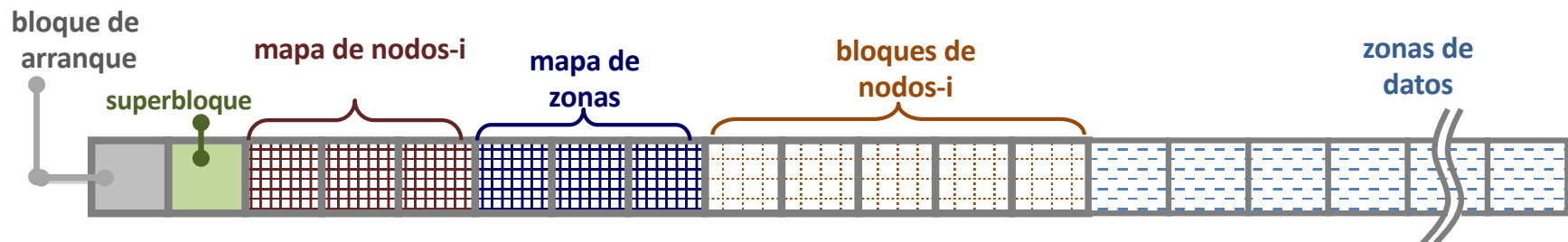
Solución:

- **Ejercicio 2: Cálculo de la estructura de una partición**

Dado un disco de 20 Mb en MINIX con los siguientes parámetros:

- Punteros a zonas de datos de 16 bits
- Tamaño de bloque 1K (1 zona = 1 bloque)
- Tamaño de nodo-i de 32 bytes
- Número máximo de nodos-i: 512

- a) Especifique claramente todas las estructuras de datos que forman el sistema de archivos y los bloques que ocupa cada una
- b) En caso de resultar dañada la estructura del mapa de bits de zonas, piense la forma de reconstruirla con la información de la que se dispone en el resto de estructuras del sistema de archivos (suponga que el resto de las estructuras están inalteradas).



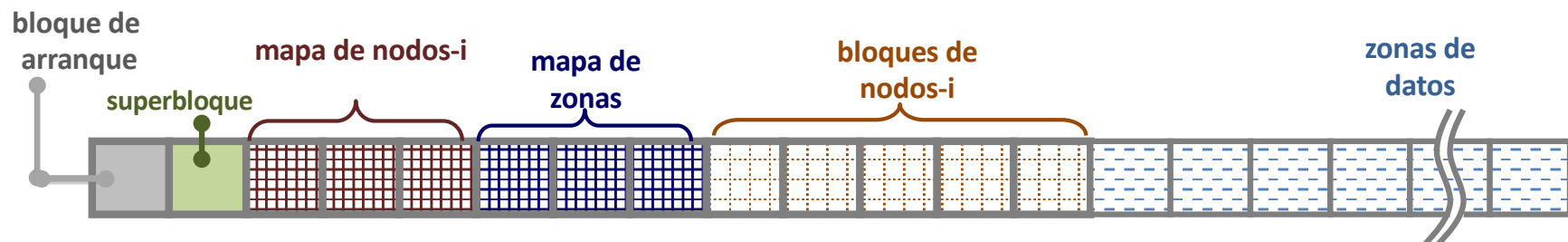
- **Ejercicio 2 (Solución): Cálculo de la estructura de una partición**

Dado un disco de **20 Mb** en MINIX con los siguientes parámetros:

- Punteros a zonas de datos de 16 bits
- Tamaño de bloque 1K (1 zona = 1 bloque)
- Tamaño de nodo-i de 32 bytes
- Número máximo de nodos-i: 512

a) Especifique claramente todas las estructuras de datos que forman el sistema de archivos y los bloques que ocupa cada una

- **Bloque de arranque = 1 Bloque**
- **Super Bloque = 1 Bloque**
- **Mapa de nodos-i → Es necesario 1 bit por cada nodo-i**
 - Número máximo de Nodos-i = 512 → 512 bits como mínimo en el Mapa de Nodos-i
 - $512 \text{ bits} / 1 \text{ bloque} = 512 \text{ Bits} / 1 \text{ Kbyte} = 512 \text{ bits} / (8 * 1024 \text{ Bits}) \rightarrow 1 \text{ Bloque para los Nodos-i}$
- **Mapa de Zonas → Es necesario 1 bit por cada zona**
 - Tamaño del disco = 20 MBytes = $20 \text{ MBytes} / 1 \text{ KBytes} = 20 \text{ K zonas} = 20 * 1024 \text{ Zonas}$
 - Son necesarios $20 * 1024 \text{ bits}$ en el mapa de zonas → $(20 * 1024 \text{ bits}) / 1 \text{ KByte} = (20 * 1024 \text{ bits}) / (8 * 1024 \text{ bits}) = 20 / 8 = 2,5 \rightarrow$ son necesarios **3 Bloques para el mapa de Zonas**
- **Nodos-i → Son necesarios 32 Bytes por cada Nodo-i**
 - $32 \text{ Bytes} * 512 \text{ Nodos-i} = 16 \text{ K Bytes} \rightarrow 16 \text{ Bloques para los Nodos-i}$



• Ejercicio 3: Búsqueda de un archivo en un directorio

Sea un sistema de archivos Minix, creado con los tamaños estándar cuyas entradas de directorio actualmente son la siguientes

1	.
1	..
3	usr
4	home
5	etc

4	.
1	..
35	prac
84	fso

3	.
1	..
60	ut12
61	sut12

84	.
4	..
90	map
91	listado

5	.
1	..

90	.
84	..

- Dibuje el árbol de directorios y archivos que corresponde a dicho sistema
- Indique de forma justificada **que números de nodos-i y cuantos bloques** es necesario acceder, si se requiere la lectura de los 128 primeros Bytes del archivo **/home/fso/listado**
- Que información deberíamos conseguir al leer los 32 primeros Bytes del archivo **/home/fso/map**

- Ejercicio 3 (solución): Búsqueda de un archivo en un directorio**

Sea un sistema de archivos Minix, creado con los tamaños estándar cuyas entradas de directorio actualmente son la siguientes

1	.
1	..
3	usr
4	home
5	etc

4	.
1	..
35	prac
84	fso

3	.
1	..
60	ut12
61	sut12

84	.
4	..
90	map
91	listado

5	.
1	..

90	.
84	..

- b) Indique de forma justificada **que números de nodos-i y cuantos bloques** es necesario acceder, si se requiere la lectura de los 128 primeros Bytes del archivo **/home/fso/listado**

/home/fso/listado

Nodo-i 1 del raíz / + Nodo-i 4 **/home** + Nodo-i 84 de **/home/fso**
+ Nodo-i 91 de **/home/fso/listado**

Total 4 nodos i (1,4,84 y 91)

• Ejercicio 3 (solución): Búsqueda de un archivo en un directorio

1	.
1	..
3	usr
4	home
5	etc

4	.
1	..
35	prac
84	fso

3	.
1	..
60	ut12
61	sut12

84	.
4	..
90	map
91	listado

5	.
1	..

90	.
84	..

b) Indique de forma justificada **que números de nodos-i y cuantos bloques** es necesario acceder, si se requiere la lectura de los 128 primeros Bytes del archivo **/home/fso/listado**

Bloques

- En un bloque de 1024 Byte pueden ubicarse 32 nodos-i de 32 bytes
 - Por tanto, los nodos del 0 al 31 se encuentran en el mismo bloque (1er bloque de nodos-i)
 - Los nodos -i del 32 al 63 se encuentran en el segundo bloque de nodos-i
 - Los nodos-i del 64 al 95 se encuentran en el tercer bloque de nodos-i
- **Bloques necesarios a consultar**
 - El bloque de la cabecera donde esta el nodo-i 1
 - El bloque del directorio raíz que contiene las entradas de directorio
 - El bloque de la cabecera que contiene el nodo-i 4
 - El bloque del directorio home que contiene la entrada fso
 - El bloque de la cabecera que contiene el nodo-i 84
 - El bloque del directorio fso que contiene la entrada listado
 - El bloque de la cabecera que contiene el nodo-i 91
 - El bloque que contiene los primeros 128 bytes del archivo listado

• Ejercicio 3: Búsqueda de un archivo en un directorio

Sea un sistema de archivos Minix, creado con los tamaños estándar cuyas entradas de directorio actualmente son la siguientes

1	.
1	..
3	usr
4	home
5	etc

4	.
1	..
35	prac
84	fso

3	.
1	..
60	ut12
61	sut12

84	.
4	..
90	map
91	listado

5	.
1	..

90	.
84	..

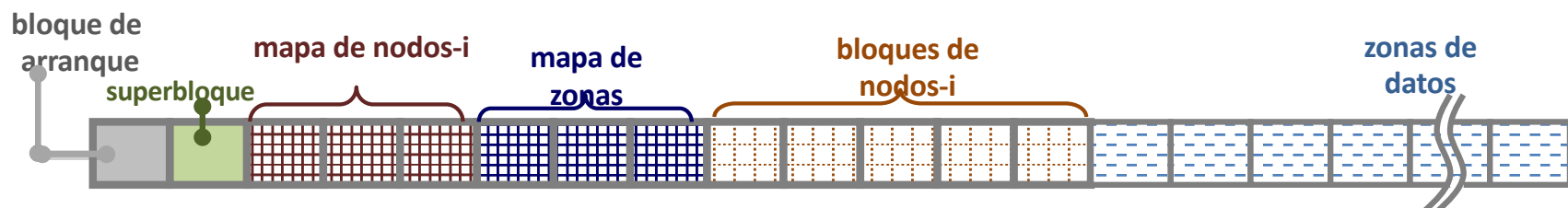
c) Que información deberíamos conseguir al leer los 32 primeros Bytes del archivo **/home/fso/map**

Dos entradas de directorio.

Dos registros con número de nodo-1 y nombre

- 90 .
- 84 ..

- Un disco con una capacidad de 8GB, se formatea con una versión de Minix. Los tamaños usados en el formateo son:
 - Tamaño de bloque = 2KBytes . Tamaño de Zona = 2^0 bloques = 1 Bloque
 - Los punteros a Zona son de 32bits=4Bytes
 - El tamaño del nodo-i es de 64 Bytes (7 punteros directos, 1 indirecto, 1 doble indirecto) .
 - Cada entrada de directorio ocupa 32 Bytes.
 - Al formatear se ha reservado espacio en la cabecera para 4.096 nodos-i
- Se pide:
 - a) Calcule el número de bloques que ocupa cada uno de los elementos de la cabecera: Mapa de bits nodos-i, Mapa de bits Zonas y Nodos-i.
 - b) Calcule el bloque que corresponde a la primera Zona de datos y el número de Zonas de datos.
 - c) Suponga que este disco almacena un único directorio, el directorio raíz que contiene 10 archivos regulares,
 - c1) Indique el número de zonas de datos que ocupa el directorio raíz
 - c2) Suponga además que cada uno de los archivos regulares contiene una información que ocupa 50KBytes e indique de forma justificada el número de zonas de datos ocupadas para este caso, tenga en cuenta tanto los datos como los metadatos del archivo.



Ejercicio 4 (solución):

- Un disco con una capacidad de 8GB, se formatea con una versión de Minix. Los tamaños usados en el formateo son:
 - Tamaño de bloque = 2KBytes . Tamaño de Zona = 2^0 bloques = 1 Bloque
 - Los punteros a Zona son de 32bits=4Bytes
 - El tamaño del nodo-i es de 64 Bytes (7 punteros directos, 1 indirecto, 1 doble indirecto) .
 - Cada entrada de directorio ocupa 32 Bytes.
 - Al formatear se ha reservado espacio en la cabecera para 4.096 nodos-i
- Se pide:
 - a) Calcule el número de bloques que ocupa cada uno de los elementos de la cabecera: Mapa de bits nodos-i, Mapa de bits Zonas y Nodos-i.

a)

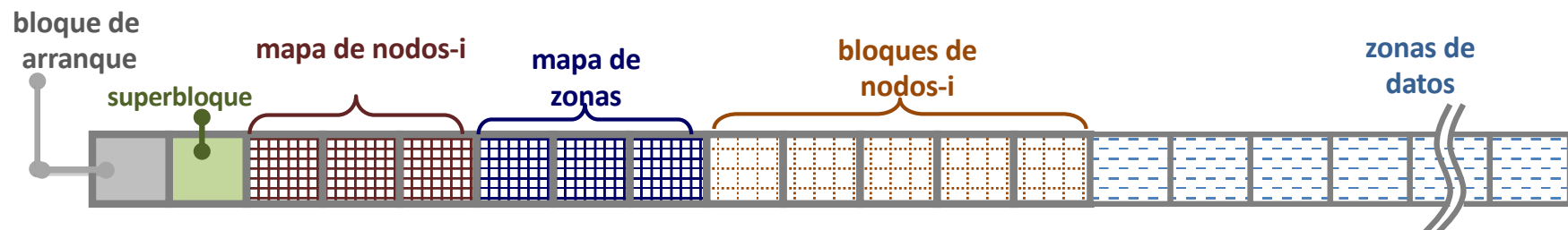
Tamaño de Bloque=2KBytes $\rightarrow 2*8* 1024$ bits

4096 nodos-i \rightarrow Mapa de nodos-i = 4096 bits $\rightarrow 1$ bloque

Nodos-i $\rightarrow 4096$ nodos-i * 64Bytes = $2^{12} * 2^6 = 2^{18}$ Bytes $\rightarrow 2^{18} / 2\text{KBytes} = 2^7 = 128$ bloques

Disco 8GBytes y Zonas = 2KBytes $\rightarrow 8\text{GBytes} / 2\text{KBytes} = 2^{22}$ zonas

Mapa de zonas con 2^{22} bits $\rightarrow 2^{22} / 2*8*1024 = 2^8$ bloques= 256 bloques

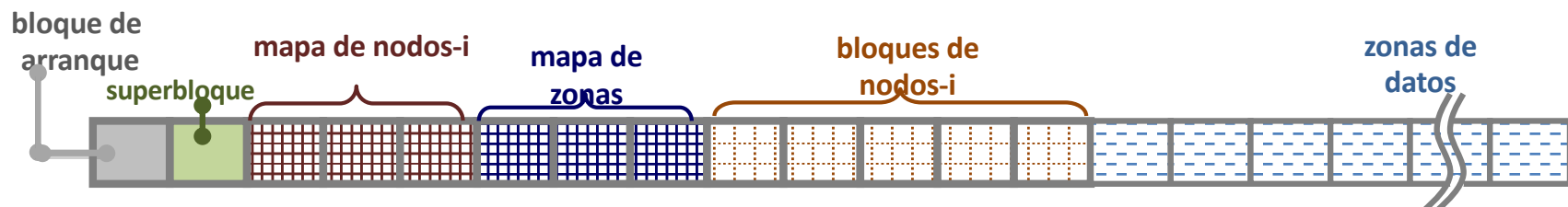


Ejercicio 4:

- Un disco con una capacidad de 8GB, se formatea con una versión de Minix. Los tamaños usados en el formateo son:
 - Tamaño de bloque = 2KBytes . Tamaño de Zona = 2^0 bloques = 1 Bloque
 - Los punteros a Zona son de 32bits=4Bytes
 - El tamaño del nodo-i es de 64 Bytes (7 punteros directos, 1 indirecto, 1 doble indirecto) .
 - Cada entrada de directorio ocupa 32 Bytes.
 - Al formatear se ha reservado espacio en la cabecera para 4.096 nodos-i
- Se pide:
 - b) Calcule el bloque que corresponde a la primera Zona de datos y el número de Zonas de datos.

b)

1 Arranque + 1 SuperBloque + 1 Mapa nodos-i + 256 Mapa de Zonas + 128 Nodos-i = 387
Bloque 387 será el primer bloque de datos = Primera zona de datos

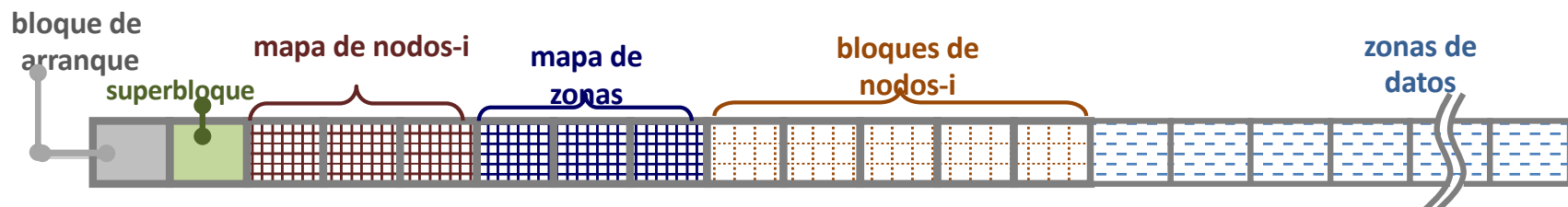


Ejercicio 4:

- Un disco con una capacidad de 8GB, se formatea con una versión de Minix. Los tamaños usados en el formateo son:
 - Tamaño de bloque = 2KBytes . Tamaño de Zona = 2^0 bloques = 1 Bloque
 - Los punteros a Zona son de 32bits = 4Bytes
 - El tamaño del nodo-i es de 64 Bytes (7 punteros directos, 1 indirecto, 1 doble indirecto) .
 - Cada entrada de directorio ocupa 32 Bytes.
 - Al formatear se ha reservado espacio en la cabecera para 4.096 nodos-i
- Se pide:
 - c) Suponga que este disco almacena un único directorio, el directorio raíz que contiene 10 archivos regulares,
 - c1) Indique el número de zonas de datos que ocupa el directorio raíz

c1)

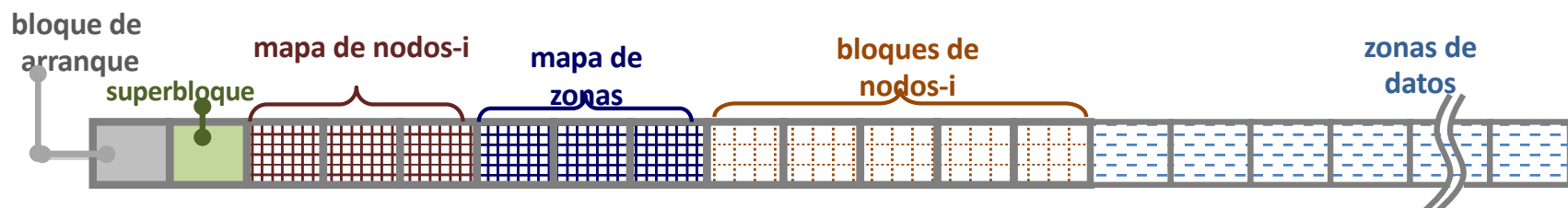
Directorio raíz contiene 10 entradas de archivos regulares + . y .. = 12 entradas →
12 entradas * 64 Bytes = 1280 Bytes → 1 zona



Ejercicio 4:

- Un disco con una capacidad de 8GB, se formatea con una versión de Minix. Los tamaños usados en el formateo son:
 - Tamaño de bloque = 2KBytes . Tamaño de Zona = 2^0 bloques = 1 Bloque
 - Los punteros a Zona son de 32bits=4Bytes
 - El tamaño del nodo-i es de 64 Bytes (7 punteros directos, 1 indirecto, 1 doble indirecto) .
 - Cada entrada de directorio ocupa 32 Bytes.
 - Al formatear se ha reservado espacio en la cabecera para 4.096 nodos-i
- c) Suponga que este disco almacena un único directorio, el directorio raíz que contiene 10 archivos regulares,
 - c1) Indique el número de zonas de datos que ocupa el directorio raíz
 - c2) Suponga además que cada uno de los archivos regulares contiene una información que ocupa 50KBytes e indique de forma justificada el número de zonas de datos ocupadas para este caso, tenga en cuenta tanto los datos como los metadatos del archivo.

50KBytes \rightarrow 25 zonas de datos \rightarrow Necesita 25 punteros a zona \rightarrow 7 punteros directos + 1 zona que contiene 18 punteros \rightarrow Total 26 zonas para cada archivos
26 zonas * 10 archivos = 260 zonas para archivos regular
260 zonas + 1 zona del raíz = 261 zonas



Dado el siguiente código en el cual se generan al menos tres procesos P1, P2 y P3:

<pre> ... pipe(fd); /*pipe1*/ pipe(fd2); /*pipe2*/ if(fork() != 0){ /**Proceso P1 ***/ dup2(fd[1], STDOUT_FILENO); close(fd[0]); close(fd[1]); dup2(fd2[0], STDIN_FILENO); close(fd2[0]); close(fd2[1]); /*tabla proceso P1*/ }else{ /**Proceso P2 ***/ dup2(fd[0], STDIN_FILENO); close(fd[0]); close(fd[1]); </pre>	<pre> pipe(fd); /*pipe3*/ if(fork() != 0){ close(fd2[0]); close(fd2[1]); dup2(fd[1], STDOUT_FILENO); close(fd[0]); close(fd[1]); /*tabla proceso P2*/ }else{ /**Proceso P3 ***/ dup2(fd[0], STDIN_FILENO); close(fd[0]); close(fd[1]); dup2(fd2[1], STDOUT_FILENO); close(fd2[0]); close(fd2[1]); /*tabla proceso P3*/ } } ... </pre>
---	---

- Indique el contenido de las tablas de descriptores de archivo para los procesos P1, P2 y P3, en los puntos del código marcados como /*tabla proceso Pi*/.
- Determine el parentesco que existe entre P1, P2 y P3 así como el esquema de redirecciones resultante de ejecutar el código.

Ejercicio-5 (solución)

Dado el siguiente código en el cual se generan al menos tres procesos P1, P2 y P3:

```
...
pipe(fd); /*pipe1*/
pipe(fd2); /*pipe2*/
if(fork() != 0){
    /**Proceso P1 ***/
    dup2(fd[1], STDOUT_FILENO);
    close(fd[0]); close(fd[1]);
    dup2(fd2[0], STDIN_FILENO);
    close(fd2[0]);
    close(fd2[1]);
    /*tabla proceso P1*/
}else{
    /**Proceso P2 ***/
    dup2(fd[0], STDIN_FILENO);
    close(fd[0]); close(fd[1]);
}
```

```
pipe(fd); /*pipe3*/
if(fork() != 0){
    close(fd2[0]);
    close(fd2[1]);
    dup2(fd[1], STDOUT_FILENO);
    close(fd[0]);
    close(fd[1]);
    /*tabla proceso P2*/
}else{
    /**Proceso P3 ***/
    dup2(fd[0], STDIN_FILENO);
    close(fd[0]);
    close(fd[1]);
    dup2(fd2[1], STDOUT_FILENO);
    close(fd2[0]);
    close(fd2[1]);
    /*tabla proceso P3*/
}
}
```

Proceso P1	
0	fd2[0]
1	fd[1]
2	stderr

/*pipe2*/
/*pipe1*/

Proceso P2	
0	fd[0]
1	fd[1]
2	stderr

/*pipe1*/
/*pipe3*/

Proceso P3	
0	fd[0]
1	fd2[1]
2	stderr

/*pipe3*/
/*pipe2*/

Ejercicio-5 (solución)

Dado el siguiente código en el cual se generan al menos tres procesos P1, P2 y P3:

<pre>... pipe(fd); /*pipe1*/ pipe(fd2); /*pipe2*/ if(fork() != 0){ /**Proceso P1 ***/ dup2(fd[1],STDOUT_FILENO); close(fd[0]); close(fd[1]); dup2(fd2[0],STDIN_FILENO); close(fd2[0]); close(fd2[1]); /*tabla proceso P1*/ }else{ /**Proceso P2 ***/ dup2(fd[0],STDIN_FILENO); close(fd[0]); close(fd[1]); }</pre>	<pre>pipe(fd); /*pipe3*/ if(fork() != 0){ close(fd2[0]); close(fd2[1]); dup2(fd[1],STDOUT_FILENO); close(fd[0]); close(fd[1]); /*tabla proceso P2*/ }else{ /**Proceso P3 ***/ dup2(fd[0],STDIN_FILENO); close(fd[0]); close(fd[1]); dup2(fd2[1],STDOUT_FILENO); close(fd2[0]); close(fd2[1]); /*tabla proceso P3*/ } }</pre>
--	--

El proceso P1 es el padre de P2, P2 es el padre de P3

Se establece un anillo de comunicación entre los tres procesos utilizando tres tubos

Parentesco



Esquema de comunicación

