



UNIVERSIDAD PÚBLICA DE NAVARRA
Grupo de Ingeniería Telemática

LABORATORIO DE TELEMÁTICA

Convocatoria ordinaria de Junio 1998

- Se permiten todo tipo de libros y apuntes
- Duración: 2 horas
- **Nota:** Responda en diferente hoja a las pregunta de Java respecto a las de UNIX

UNIX

Pregunta 1.- Mencione los tres tipos de configuración del buffer en un stream y qué condiciones deben cumplirse en cada uno de los casos para que el buffer se vuelque al destino.

Buffer completo: Los volcados se realizan al llenarse el buffer.

Buffer a líneas: El volcado se realiza al llenarse o al encontrar un \n.

Sin Buffer: El volcado se realiza inmediatamente, como si no hubiera buffer.

En todos los casos también se vuelca ante un fflush, un fclose, un exit o llegar al final de la función main

Pregunta 2.- Dado el siguiente código de programa, que compila correctamente:

```
[comienzo de programa]
/* #include omitidos en el enunciado por acortarlo */
main()
{
int fd;

fd=open("prueba", O_CREAT|O_WRONLY, 0600);
printf("mensaje 1\n");
dup2(fd, STDOUT_FILENO);
printf("mensaje 2\n");
}
[final de programa]
```

¿Cuáles serán las salidas por pantalla y el contenido del fichero "prueba" supuesto que ninguna función da error?

Salida por pantalla:

mensaje 1

Contenido del fichero:

mensaje 2

Pregunta 3.- Dado el siguiente código de programa, que compila correctamente:

```
[comienzo de programa]
/* #include omitidos en el enunciado por acortarlo */
main()
{
int fd;

fd=open("prueba", O_CREAT|O_WRONLY, 0600);
printf("mensaje 1");
dup2(fd, STDOUT_FILENO);
```

```
printf("mensaje 2\n");
}
[final de programa]
```

Se ejecuta. No se produce salida por pantalla y el contenido del fichero "prueba" pasa a ser:

```
[comienzo del fichero prueba]
mensaje 1mensaje 2
[final del fichero prueba]
```

Explique cómo y por qué ha afectado el \n que se ha quitado respecto a la pregunta anterior.

Nota: El stream `stdout` tiene un buffer de tamaño 8192 bytes, configurado a líneas.

Al hacer el primer printf el texto se coloca en el buffer de `stdout`, pero no se vuelve por no haberse llenado ni encontrado un \n. Con el `dup2` se redirige la salida estándar al fichero. El siguiente printf incluye el \n luego fuerza el volcado del contenido del buffer (que es `mensaje 1mensaje2\n`) que se realiza al fichero en vez de al posible terminar como efecto de la redirección.

Pregunta 4- Marque las funciones de entre las siguientes que devuelven un descriptor de fichero:

a) open c) `msgget` e) `read`
b) `fopen` d) dup f) `write`

Pregunta 5.- ¿De cuántos descriptores de fichero consta una pipe? ¿Con qué llamada al sistema se obtiene una nueva pipe?

Consta de dos. La llamada es `pipe()`.

Pregunta 6.- El siguiente programa:

```
[comienzo de programa]
/* #include omitidos en el enunciado por acortarlo */
main()
{
    int    valor=0;
    pid_t  pid;

    pid=fork();
    if (pid==0) {
        valor=1;
        printf("valor (proceso 1) = %i\n", valor);
    }
    else if (pid>0)
        printf("valor (proceso 2) = %i\n", valor);
    else printf("Error de inicializacion\n");

    exit(0);
}
[final de programa]
```

Suponiendo que el `fork()` no devuelve error, dé la salida por pantalla que provoca el programa en una posible ejecución.

Un posible resultado por pantalla es:

```
valor (proceso 1) = 1
valor (proceso 2) = 0
```

Otro posible es:

```
valor (proceso 2) = 0
valor (proceso 1) = 1
```

Pregunta 7.- El siguiente programa:

```
[comienzo de programa]
/* #include omitidos en el enunciado por acortarlo */
main()
{
    int    valor=0;
    pid_t  pid;

    pid=fork();
    if (pid<0) exit(-1);
    else printf("inicializacion correcta\n");

    if (pid==0) {
        valor=1;
        printf("valor (proceso 1) = %i\n", valor);
    }
    else if (pid>0)
        printf("valor (proceso 2) = %i\n", valor);

    exit(0);
}
[final de programa]
```

Suponiendo que el `fork()` no devuelve error, dé la salida por pantalla que provoca el programa en una posible ejecución.

Un posible resultado por pantalla es:

```
inicializacion correcta
valor (proceso 1) = 1
inicializacion correcta
valor (proceso 2) = 0
```

Pregunta 8.- ¿Qué posibles valores devuelve la llamada del sistema `fork()`? ¿Qué significado tienen?

Devuelve -1 si no se ha podido ejecutar con éxito, 0 al nuevo proceso hijo y un número > 0 al padre, que es el PID del nuevo proceso.

Pregunta 9.- ¿Con qué llamada del sistema se puede enviar una señal a un proceso?

`kill`

Pregunta 10.- Sea un programa que contenga el siguiente código:

```
[comienzo de trozo de programa]
msgrcv cola, (void *)&msg, MAX_MSG, 0, 0);
[final de trozo de programa]
```

Al ejecutarlo, `cola` hace referencia a una cola de mensajes existente y de la cual el proceso puede leer.

¿Qué sucede si no existe ningún mensaje en la cola al llegar la ejecución al `msgrcv()`? (Solo una respuesta es correcta)

- a) El programa se queda bloqueado en el `msgrcv()` hasta que llegue uno
- b) `msgrcv()` devuelve error
- c) El programa da un fallo de segmentación al encontrar la cola vacía

Pregunta 11.- ¿Qué comando hay que usar para compilar la clase matriz?

d) `javac matriz.java`

Pregunta 12.- En un método de otra clase tenemos declarada la variable:

matriz A;

¿Cómo se construye un objeto de tipo matriz para guardar una matriz 3x2?

b) `A = new matriz(3,2);`

Pregunta 13.- Programe un método para la clase matriz que permita acceder a los datos de la matriz sin tener que hacer `A.elemento[i][j]`. En su lugar se desea hacer `A.getElemento(i,j)` para ver el valor de un elemento y `A.setElemento(i,j,x)` para poner el elemento i,j al valor x;

```
public void setElemento(int i,int j,float x) {
    elemento[i][j] = x;
}

public float getElemento(int i,int j) {
    return elemento[i][j];
}
```

Pregunta 14.- Programe un método que permita comparar una matriz con otra. El método debe permitirnos comparar las matrices A y B utilizando la expresión `A.equals(B)` que devuelve true si son iguales y false si no. (Nota: se supone que dos matrices son iguales si todos sus elementos son iguales)

```
public boolean equals(matriz B) {
    int i,j;
    boolean iguales=true;

    if ( (elemento.length!=B.elemento.length)
        ||(elemento[0].length!=B.elemento[0].length) ) {

        return false;
    }

    for (i=0;i<elemento.length;i++) {
        for (j=0;j<elemento[0].length;j++) {
            if (elemento[i][j]!=B.elemento[i][j]) {
                iguales=false;
            }
        }
    }
    return iguales;
}
```

Pregunta 15.- ¿Qué modificación hay que hacer en el método anterior para que escriba cada 10 segundos?

Cambiar `Thread.sleep(2000)` por `Thread.sleep(10000)` ya que el argumento esta en milisegundos.

Pregunta 16.- ¿Y para que escriba el tiempo que lleva esperando? con este formato:

Han pasado 10 segundos

Han pasado 20 segundos

Han pasado 30 segundos

Cambiar el `System.out.println("zzz");` *por*
`System.out.println("Han pasado "+(i*10)+" segundos");`

Pregunta 17.- ¿Qué diferencia fundamental hay entre un Canvas y un Panel?

La clase Panel es subclase de Container y la clase Canvas solo es subclase de Component por lo que los Panels pueden contener otros componentes mientras que los Canvas no.

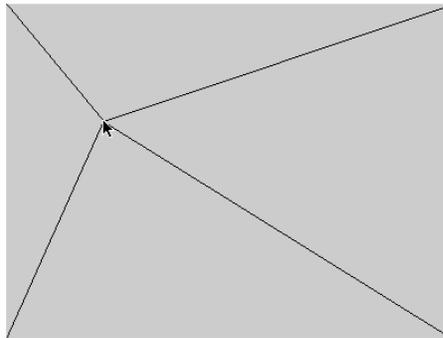
Pregunta 18.- Supongamos que se escribe un programa en Java que incluye la línea
`if (x=0) System.out.println("X vale 0!!!");`

¿Qué sucede cuando se intenta compilarlo y ejecutarlo?

c) Da error de compilación porque $x=0$ no es un booleano.

Pregunta 19.- ¿Qué hace este Xcanvas? Explique el comportamiento y dibuje lo que aparece en el appletviewer (o en el navegador de Web)

El Xcanvas dibuja líneas desde sus 4 esquinas hasta la posición del ratón si el puntero está sobre el Xcanvas. En caso de que el puntero no esté sobre el Xcanvas no dibuja nada porque se desactiva el booleano on.



Pregunta 20.- Escriba el método `init()` de un Applet que muestre 2 veces el Xcanvas uno al lado del otro.

```
public void init() {  
    Xcanvas x1,x2;    // Se podría suponer que están declarados  
                    // fuera del método  
  
    x1 = new Xcanvas();  
    x2 = new Xcanvas();  
  
    setLayout(new GridLayout(1,2));  
    add(x1);  
    add(x2);  
}
```