

Comenzado el	lunes, 1 de junio de 2020, 09:48
Estado	Finalizado
Finalizado en	lunes, 1 de junio de 2020, 10:06
Tiempo empleado	18 minutos
La puntuación	6,25/15,00
Calificación	4,17 de 10,00 (42%)

Pregunta 1


Correcta

Puntúa 1,00 sobre 1,00

🚩 Marcar pregunta

¿Cuál de las siguientes llamadas a la función waitpid() no bloquea al proceso que la ejecuta?

Selecciona una:

- ☐ a. mypid = waitpid(pid, &status, WUNTRACED);
- ☐ b. Ninguna de las otras tres respuestas es correcta.
- ☐ c. mypid = waitpid(pid, &status, 0);
- ☒ d. mypid = waitpid(pid, &status, WUNTRACED | WNOHANG); 

La respuesta correcta es: mypid = waitpid(pid, &status, WUNTRACED | WNOHANG);

Pregunta 2


Incorrecta

Puntúa -0,25 sobre 1,00

🚩 Marcar pregunta

En un Sistema Operativo Unix, un proceso ya finalizado cuya señal SIGCHLD no ha sido aún atendida

Selecciona una:

- ☐ a. Permanece en estado "zombie".
- ☐ b. Es expulsado del sistema inmediatamente.
- ☐ c. Es adoptado por el proceso "init".
- ☒ d. Será resucitado por el padre. 

La respuesta correcta es: Permanece en estado "zombie".

Pregunta 3


Incorrecta

Puntúa -0,25 sobre 1,00

🚩 Marcar pregunta

¿Qué valores de las variables `status` y `ground` guardan una mayor relación? (esto es, un proceso tiene una mayor probabilidad de coincidir en ellos)

Selecciona una:

- ☐ a. status=SUSPENDIDO y ground=DETENIDO.
- ☐ b. status=FINALIZADO y ground=DETENIDO.
- ☒ c. status=SUSPENDIDO y ground=SEGUNDOPLANO. 
- ☐ d. status=REANUDADO y ground=PRIMERPLANO.

La respuesta correcta es: status=SUSPENDIDO y ground=DETENIDO.

Pregunta 4


Si quiero cambiar el aspecto de mi Shell para la recepción de comandos, "COMANDO->", ¿Qué parte del código debo cambiar?

Correcta

Puntúa 1,00 sobre 1,00

 Marcar pregunta

Selecciona una:

- ☐ a. El proceso padre.
- ☐ b. El manejador de señales.
- ☐ c. El proceso hijo.
- ☒ d. El primer bucle while que hay dentro del main() 

La respuesta correcta es: El primer bucle while que hay dentro del main()

Pregunta 5


Correcta

Puntúa 1,00 sobre 1,00

 Marcar pregunta

¿Puede un proceso zombie que ha terminado irregularmente en tu Shell reanudar su ejecución?

Selecciona una:

- ☒ a. No. 
- ☐ b. Sí, siempre que el manejador de señales lo resucite enviándole una señal.
- ☐ c. Sí, siempre que el padre o el manejador de señales lo resucite enviándole una señal.
- ☐ d. Sí, siempre que el padre lo resucite enviándole una señal.

La respuesta correcta es: No.

Pregunta 6


Correcta

Puntúa 1,00 sobre 1,00

 Marcar pregunta

Fíjate en los 3 argumentos de las llamadas a `waitpid(pgid, &status, flags)`; ¿por qué `pgid` y `flags` se pasan por valor y `status` se pasa por referencia?

Selecciona una:

- ☐ a. Porque `waitpid()` es sensible a multitud de señales.
- ☒ b. Porque `waitpid()` devuelve información en la variable `status`, mientras que los otros dos parámetros le suministran información. 
- ☐ c. Porque `waitpid()` puede recoger señales además de esperar a la finalización de un hijo.
- ☐ d. Porque de no ser así, se bloquea la llamada a `waitpid()`, y no nos interesa ese comportamiento para nuestro Shell.

La respuesta correcta es: Porque `waitpid()` devuelve información en la variable `status`, mientras que los otros dos parámetros le suministran información.

Pregunta 7


Incorrecta

Puntúa -0,25 sobre 1,00

 Marcar pregunta

En la primera fase de nuestro proyecto Shell, si pulsamos desde el teclado CTRL+C se mata tanto al proceso hijo lanzado por el Shell como a éste último. ¿Qué debemos hacer en las fases siguientes para que esto no suceda, y que las señales asociadas a las combinaciones de teclas como CTRL+Z y CTRL+C le lleguen sólo al proceso lanzado en primer plano?

Selecciona una:

- ☐ a. Independizar al nuevo proceso en su propio grupo de procesos y asignarle el terminal.
- ☐ b. Instalar un manejador para la señal SIGCHLD.
- ☒ c. Utilizar la macro `ignore_terminal_signals()` proporcionada en las tareas auxiliares. 
- ☐ d. Utilizar la opción `WUNTRACED` en la llamada a `waitpid()` correspondiente.

La respuesta correcta es: Independizar al nuevo proceso en su propio grupo de procesos y asignarle el terminal.

Pregunta 8


Correcta

Puntúa 1,00 sobre 1,00

 Marcar pregunta

¿Por qué el manejador de señales debe recorrer la lista de procesos íntegra en lugar de terminar cuando encuentra el primer proceso que ha cambiado de estado?

Selecciona una:

- ☐ a. Porque un mismo proceso puede emitir múltiples señales.
- ☐ b. No es necesario recorrer la lista de procesos íntegra.
- ☒ c. Porque hay ocasiones en que nuestro Shell bloquea las señales con `block_SIGCHLD()` y luego las desbloquea con `unblock_SIGCHLD()`, y durante ese espacio de tiempo se pueden suceder múltiples señales. 
- ☐ d. Porque nuestro manejador se activa desde el proceso padre, que puede ser a su vez emisor de señales.

La respuesta correcta es: Porque hay ocasiones en que nuestro Shell bloquea las señales con `block_SIGCHLD()` y luego las desbloquea con `unblock_SIGCHLD()`, y durante ese espacio de tiempo se pueden suceder múltiples señales.

Pregunta 9


Incorrecta

Puntúa -0,25 sobre 1,00

 Marcar pregunta

Si quiero imprimir en pantalla un mensaje que diga "Señal recibida" cada vez que se emita una señal `SIGCHLD`, ¿qué parte del código del shell debo cambiar?

Selecciona una:

- ☐ a. El código del proceso padre tras el `fork()`.
- ☒ b. El código del proceso hijo tras el `fork()`. 
- ☐ c. El código del `main()`, entre su inicio y la llamada a `fork()`.
- ☐ d. El código del manejador de señales.

La respuesta correcta es: El código del manejador de señales.

Pregunta 10


Correcta

Puntúa 1,00 sobre 1,00

 Marcar pregunta

¿Cuál es la diferencia entre el comando `jobs` y el comando `ps`?

Selecciona una:

- ☐ a. `ps` lista procesos en primer plano y `jobs` lista procesos en segundo plano.
- ☐ b. `ps` lista directorios y `jobs` procesos.
- ☒ c. `jobs` lista sólo trabajos lanzados desde el shell y `ps` lista procesos en general. 
- ☐ d. Son comandos equivalentes.

La respuesta correcta es: `jobs` lista sólo trabajos lanzados desde el shell y `ps` lista procesos en general.

Pregunta 11


Incorrecta

En la implementación de tu Shell, ¿qué parte(s) del código utiliza(n) la llamada a `killpg()` para enviar una señal de reanudación a un proceso detenido?

Puntúa -0,25 sobre 1,00

 Marcar pregunta

Selecciona una:

- ☐ a. El proceso hijo tras la llamada a fork().
- ☐ b. El manejador de señales.
- ☒ c. El proceso padre tras la llamada a fork(). 
- ☐ d. La implementación de los comandos fg y bg.

La respuesta correcta es: La implementación de los comandos fg y bg.

Pregunta 12


Incorrecta

Puntúa -0,25 sobre 1,00

 Marcar pregunta

En la implementación de las diversas fases de tu Shell, hay una fase que no requiere modificar el código del proceso padre. ¿Cuál?

Selecciona una:

- ☒ a. La cuarta. 
- ☐ b. La quinta.
- ☐ c. La primera.
- ☐ d. La tercera.

La respuesta correcta es: La quinta.

Pregunta 13


Correcta

Puntúa 1,00 sobre 1,00

 Marcar pregunta

¿Qué comando interno de entre los que se han solicitado para tu Shell resulta más difícil de implementar?

Selecciona una:

- ☐ a. logout
- ☒ b. fg 
- ☐ c. jobs
- ☐ d. cd

La respuesta correcta es: fg

Pregunta 14

Incorrecta

Puntúa -0,25 sobre 1,00

 Marcar pregunta

Queremos implementar un comando **silencio** en nuestro Shell que permita deshabilitar de forma temporal la entrada de comandos. El Shell ignorará cualquier comando que introduzca el usuario desde que teclea "silencio on" hasta que restaura el funcionamiento normal tecleando "silencio off". Ejemplo:

```
COMANDO->silencio on
```

```
COMANDO->pwd      (no haría nada)
```

```
COMANDO->ls        (no haría nada)
```

```
COMANDO->silencio off
```

```
COMANDO->pwd      (continúa con su comportamiento normal)
```

```
/home/usuario
```

```
COMANDO->
```

¿En qué parte del código de tu Shell desarrollarías la implementación para ese comando **silencio**?

Selecciona una:

- ☒ a. En el proceso padre tras la llamada a fork(). ❌
- ☐ b. En el código del main(), entre su inicio y la llamada a fork().
- ☐ c. En el manejador de señales.
- ☐ d. En el proceso hijo tras la llamada a fork().

La respuesta correcta es: En el código del main(), entre su inicio y la llamada a fork().

Pregunta 15

Correcta

Puntúa 1,00 sobre 1,00

🚩 Marcar pregunta

Los comandos internos del shell son aquellos interpretados por el shell y que no corresponden a comandos ejecutables. Entre ellos, se encontraría

Selecciona una:

- ☒ a. bg ✅
- ☐ b. ps
- ☐ c. ls
- ☐ d. wc

La respuesta correcta es: bg

Finalizar revisión



Universidad de Málaga · Avda. Cervantes, 2. 29071 MÁLAGA · Tel. 952131000 · info@uma.es

© Todos los derechos reservados

Comenzado el	miércoles, 6 de junio de 2018, 20:35
Estado	Finalizado
Finalizado en	miércoles, 6 de junio de 2018, 20:36
Tiempo empleado	1 minuto 6 s
La puntuación	13,25/14,50
Calificación	9,14 de 10,00 (91%)

Pregunta 1

Correcta

Puntúa 1,00 sobre 1,00



Marcar pregunta

Un ejecutable compilado con la opción `-g`, se caracteriza fundamentalmente por:

Seleccione una:

- ☒ A. Ser un ejecutable con información de depuración (para gdb por ejemplo) ✓
- ☐ B. Ser un ejecutable con máxima optimización
- ☐ C. Ser un ejecutable generado estáticamente
- ☐ D. Ser un ejecutable con optimización media

Pregunta 2

Correcta

Puntúa 1,00 sobre 1,00



Marcar pregunta

¿Qué proceso lidera un grupo de procesos de terminal?

Seleccione una:

- ☐ a. Aquél que llegó primero al grupo
- ☐ b. Aquél que se ejecute a mayor velocidad
- ☒ c. Aquél cuyo pid coincida con el identificador del grupo ✓
- ☐ d. Aquél cuyo padre tenga mayor prioridad (nice)

Pregunta 3

Correcta

Puntúa 1,00 sobre 1,00



Marcar pregunta

¿Cuál de las siguientes llamadas a la función `waitpid` no bloquea al proceso llamante?

Seleccione una:

- ☐ A. `wpid = waitpid (pid, &status, WUNTRACED);`
- ☐ B. `wpid = waitpid (pid, &status, 0);`
- ☒ C. `wpid = waitpid (pid, &status, WUNTRACED|WNOHANG);` ✓
- ☐ D. `wpid = waitpid (pid, &status, WNONBLOCKING);`

Pregunta 4

Correcta

Puntúa 1,00 sobre 1,00



Marcar pregunta

Cuando en nuestro shell ejecutamos:

```
> cat &
```

Seleccione una:

- ☒ a. Como hemos lanzado a ejecutar cat en segundo plano e intenta leer del terminal, se suspende. ✓
- ☐ b. El proceso creado para la ejecución del cat finalizará de forma abrupta.
- ☐ c. Cat obtiene el terminal, ya que lo utiliza como entrada estándar
- ☐ d. Cat comienza a ejecutarse en primer plano



Pregunta 5

Correcta

Puntúa 1,00 sobre 1,00

 Marcar pregunta

¿Cuáles de las siguientes señales se consideran "señales de terminal" ignoradas por el proceso shell?

Seleccione una o más de una:

- ☒ a. SIGTTIN ✓
- ☐ b. SIGCONT
- ☒ c. SIGTTOU ✓
- ☒ d. SIGSTOP ✓
- ☐ e. SIGCHLD
- ☐ f. SIGABRT
- ☒ g. SIGINT ✓

Pregunta 6

Correcta

Puntúa 1,50 sobre 1,50

 Marcar pregunta

En lenguaje C, se quiere replicar una cadena "cmd" que contiene un comando, ¿qué opciones de las siguientes son correctas?

Seleccione una o más de una:

- ☐ a.

```
char *copia;  
strcpy(copia, cmd);
```
- ☒ b.

```
char copia[256];  
strcpy(copia, cmd);
```

 ✓
- ☒ c.

```
char *copia;  
copia = strdup(cmd);
```

 ✓
- ☐ d.

```
char copia[256];  
copia = strdup(cmd);
```
- ☐ e.

```
char *copia[256];  
strcpy(copia, cmd);
```

Pregunta 7

Correcta

Deseamos compilar el programa "programa.c":

```
#include "math.h"  
main(){ ... }
```

Si math.h está en el directorio /usr/local/include, debermos compilar como:

WUOLAH

Puntúa 1,00 sobre 1,00

🚩 Marcar pregunta

Seleccione una:

- ☐ A. gcc -L/usr/local/include programa.c
- ☒ B. gcc -I/usr/local/include programa.c ✓
- ☐ C. gcc -lmath programa.c
- ☐ D. gcc -g programa.c

Pregunta 8

Correcta

Puntúa 1,00 sobre 1,00

🚩 Marcar pregunta

Suponiendo que ya se ha ejecutado el siguiente fragmento de un programa en C, selecciona las opciones correctas:

```
#include <stdio.h>

int main(int argc, char *argv[]) {
    int v[3]={7,8,9};
    int *p=0;
    p=v;
    ...
}
```

Seleccione una o más de una:

- ☐ A. Con printf("%d,%d,%d\n",p[0], p[1], p[2]); en pantalla obtendríamos: 9, 8, 7
- ☐ B. Con printf("%p\n", &p) y con printf("%p\n", &v) se imprime el mismo valor por pantalla
- ☒ C. Con printf("%p\n",p) y con printf("%p\n",&v[0]) se imprime el mismo valor por pantalla ✓
- ☒ D. Con printf("%p\n",p) y con printf("%p\n",v) se imprime el mismo valor por pantalla ✓

Pregunta 9

Correcta

Puntúa 1,00 sobre 1,00

🚩 Marcar pregunta

En el contexto de un sistema Unix, ¿qué pareja de términos, relacionados con el estado de un job, no encajan?

Seleccione una:

- ☐ a. STOPPED - SUSPENDED
- ☒ b. SLEEP - SUSPEND ✓
- ☐ c. RUNNING - RUNNABLE
- ☐ d. BLOCK - SLEEP

Pregunta 10

Incorrecta

Puntúa -0,25 sobre 1,00

🚩 Marcar pregunta

Cuando se pulsa la combinación de teclas Control+C se envía una señal para detener todos los procesos del grupo de procesos en primer plano. ¿quién envía esta señal SIGINT?

Seleccione una:

- ☐ A. El shell
- ☒ B. El líder de la sesión al que pertenece el terminal ✗
- ☐ C. El proceso en ejecución dentro del grupo de procesos de primer plano
- ☐ D. El driver del Terminal

Pregunta 11

Correcta

Puntúa 1,00 sobre 1,00

¿Por qué bloquearemos la señal SIGCHLD al modificar la lista de jobs de un shell?

Seleccione una:

- ☐ a. Porque la lista suele estar ubicada en la pila del programa (stack)

WUOLAH

- ☒ b. Porque la señal es una acción asíncrona que puede modificar a su vez la lista ✓
- ☐ c. Porque así lo define el estándar POSIX
- ☐ d. Porque la lista no suele ser una variable compartida

Pregunta 12

Correcta

Puntúa 1,00 sobre 1,00

Ejecutando la línea de compilación gcc -O3 helloworld.c

Seleccione una:

- ☐ A. se generará un ejecutable 03.out si no hay errores
- ☐ B. no sabemos el nombre del fichero ejecutable creado
- ☒ C. se generará un ejecutable denominado a.out si no hay errores ✓
- ☐ D. se generará un ejecutable denominado helloworld si no hay errores

Pregunta 13

Correcta

Puntúa 1,00 sobre 1,00

En el código de un shell con control de procesos encontramos que las siguientes sentencias de código se ejecutan tanto en el shell como en los hijos individuales:

```
...
setpgid (pid, pgid);
if (foreground)
    tcsetpgrp (shell_terminal, pgid);
...
```

Ello es debido:

Seleccione una:

- ☐ a. A que el shell tiene mayor prioridad de planificación
- ☐ b. A que el proceso padre debe recuperar el terminal siempre que un hijo termine
- ☒ c. A las posibles condiciones de carrera causadas por la concurrencia entre procesos ✓
- ☐ d. A que no se está utilizando un planificador de tiempo compartido

Pregunta 14

Correcta

Puntúa 1,00 sobre 1,00

La acción sigblock(...) sobre una señal...

Seleccione una:

- ☐ a. ... restablece el comportamiento por defecto de la señal SIGBLOCK
- ☐ b. ... bloquea el proceso cuando recibe la señal en cuestión
- ☒ c. ... bloquea la señal, de manera que cuando llega queda pendiente ✓
- ☐ d. ... provoca que se ignore dicha señal

Finalizar revisión