

(1) Dada la siguiente salida en una terminal linux:

```
usuario@maquina ~ $ ps af
  PID TTY          STAT       TIME COMMAND
 2451 pts/0        Ss           0:00   bash
 2511 pts/0        T            0:00   \_ vi
 2513 pts/0        R+           0:00   \_ ps af
 1323 tty1        Ss+          0:00   /sbin/getty -8 38400 tty1
 1176 tty6        Ss+          0:00   /sbin/getty -8 38400 tty6
 1171 tty3        Ss+          0:00   /sbin/getty -8 38400 tty3
 1170 tty2        Ss+          0:00   /sbin/getty -8 38400 tty2
 1164 tty5        Ss+          0:00   /sbin/getty -8 38400 tty5
 1161 tty4        Ss+          0:00   /sbin/getty -8 38400 tty4
```

¿Que representan los diferentes valores de la columna STAT? ¿Por qué se genera un árbol con los procesos “vi” y “ps”?

(2) Cuando se ejecuta el siguiente código.

```
for i:=1 to 3 do begin
  if fork <> 0 then begin
    Write('a');
    if fork = 0 then begin
      Write('c');
    else begin
      Write('d');
      exit(0);
    end;
  else begin
    Write('b');
  end;
end.
```

¿Cuántas veces se muestra cada letra? (Utilice como ayuda una prueba de escritorio con la salida de cada proceso).

(3) Para cada una de las siguientes llamadas al sistema:

- **fork** (**fpFork**)
- **exec** (**fpExecv**)
- **kill** (**fpKill**)

De al menos una condición por la cual pueden devolver error.

(4) En un sistema GNU/Linux, un directorio contiene los siguientes archivos:

Archivo	Dueño	Grupo	Permisos
prog	juan	alumnos	rwX--X--
datos	juan	alumnos	rw-rw-r--

El archivo prog contiene una aplicación que utiliza el archivo datos como base de datos.

¿Qué ocurrirá si el usuario ana ejecuta **prog**?

¿Hay alguna forma de asegurar que pueda usar prog sin modificar los permisos de datos?