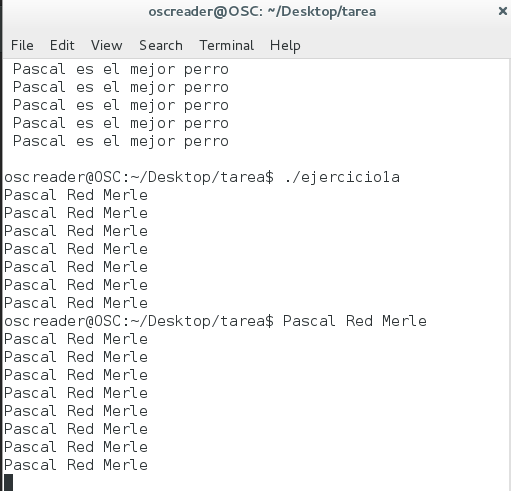
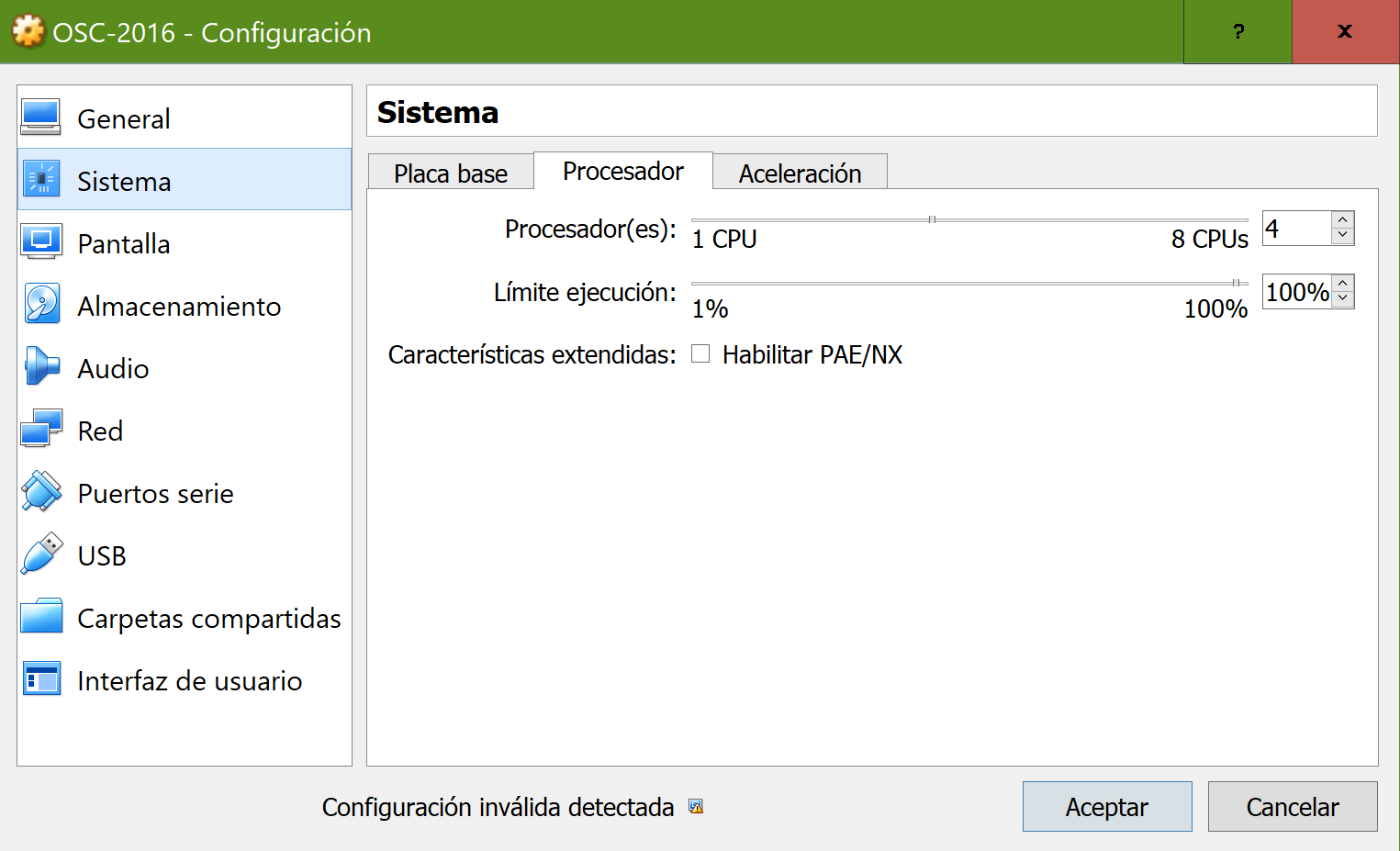
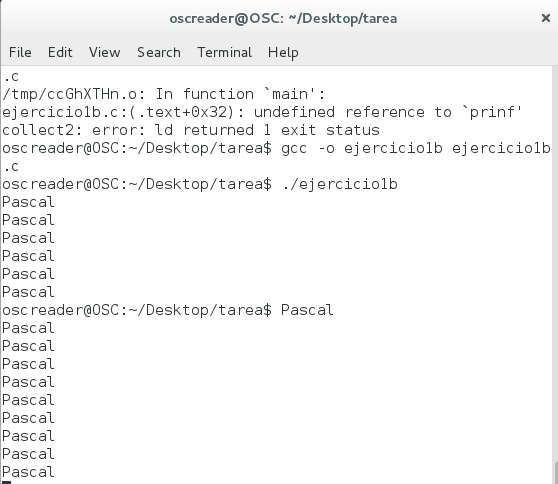
Universidad del Valle de Guatemala

Sistemas Operativos

Ana lucia Diaz Leppe #151378

**Laboratorio 3**

**Ejercicio 1:**

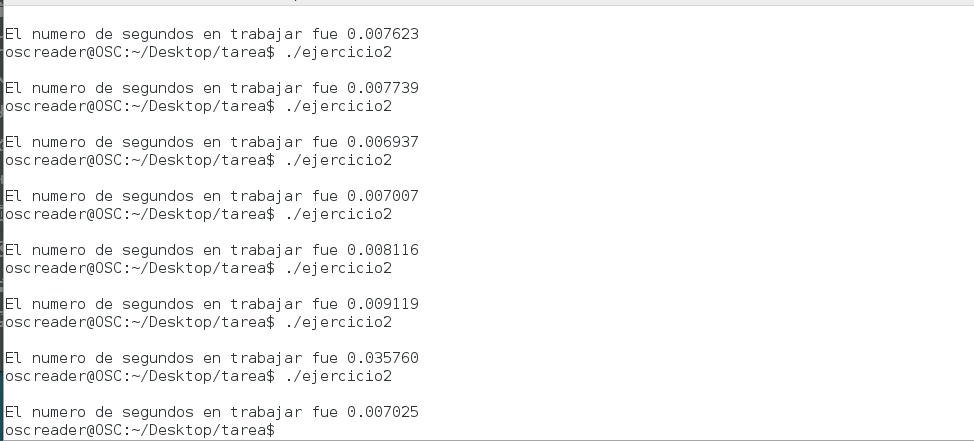
* **¿Cuántos procesos se crean en cada uno de los programas?**

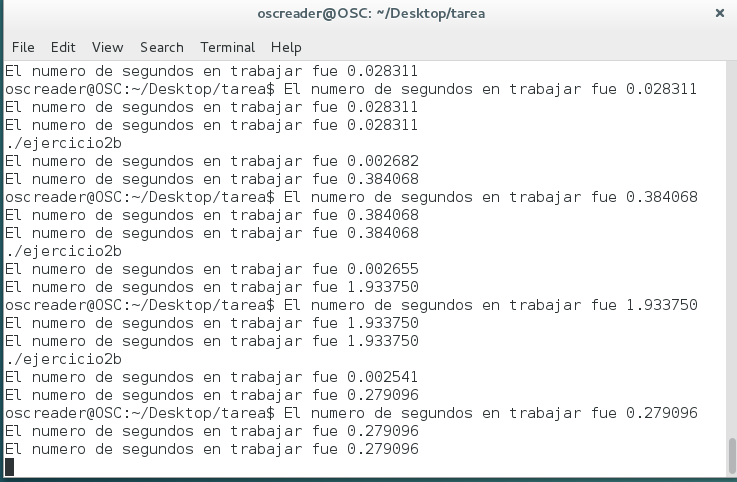
En el primer programa se utiliza 2 elevado a la n, es decir 2^4 por lo tanto se imprime 16 veces. En el segundo caso se ejecutan 2^4-1. Si sumamos todos los niveles del árbol anterior para i = 0 a n-1, obtenemos 2n - 1. Por lo tanto, habrá 2n - 1 procesos secundarios.

* **Explique cómo se crea cada proceso y qué sucede después.**

Un sistema fork llama procesos de generación como hojas de un árbol binario de crecimiento. Como llamamos el fork cuatro veces se reprodujo 2^4 = 16 procesos. Todos estos 16 procesos forman la hoja de hijos del árbol binario. Por lo tanto, Pascal Red Merle se imprimió esa cantidad de veces. En el segundo caso, se ejecutaron 2^4 – 1 debido a que: se crea un hijo en el proceso que se genera el primer fork, esto da en consecuencia 2 diferentes hijos del proceso creados por el segundo fork y esto pasara consecutivamente. Por lo tanto tenemos el caso de 2^4-1.

**Ejercicio 2**





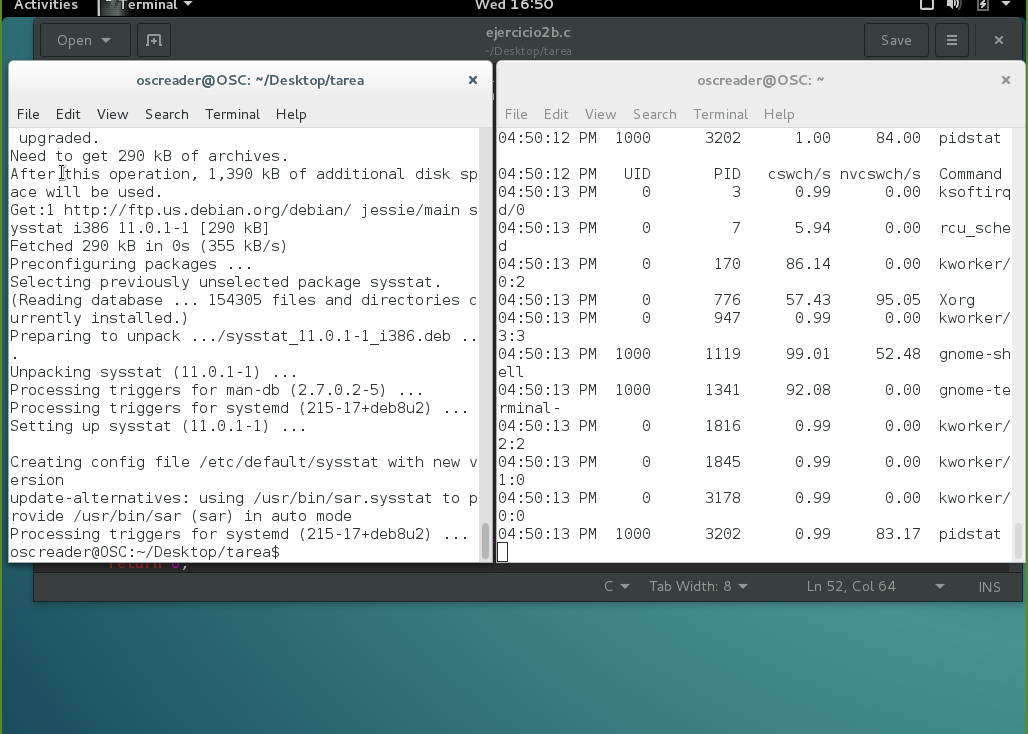
* **¿Cuál, en general, toma tiempos más largos?**

Se tarda mas el de los procesos, es decir la opción B.

* **¿Qué causa la diferencia de tiempo, o por qué se tarda más el que se tarda más?**

El proceso padre al crear hijos puede darse 2 posibilidades en términos de ejecución. El proceso padre se ejecuta con los procesos hijos de manera concurrente. El proceso padre espera a que todos los procesos hijos terminen, Es debido a esto que se puede llegar a tomar mas tiempo. Además utilizamos un wait(NULL) el cual espera hasta que el proceso del hijo ya haya terminado.

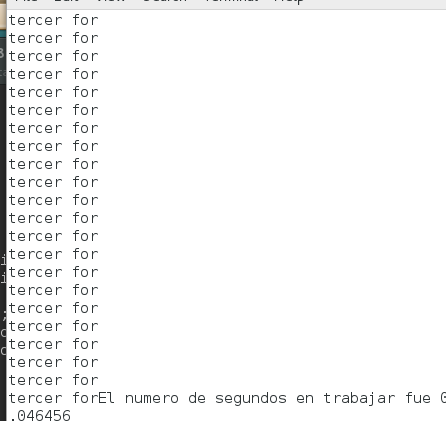
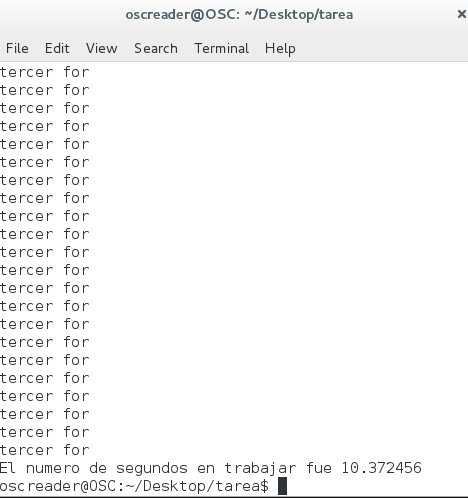
**Ejercicio 3**

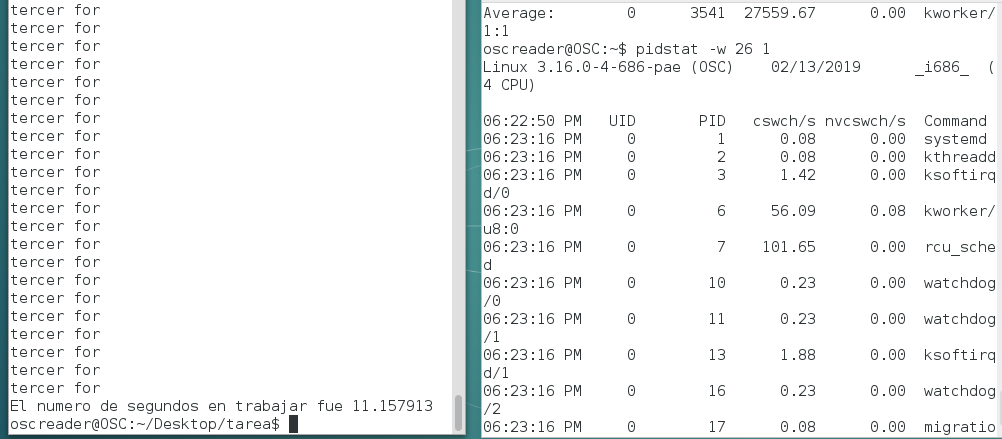


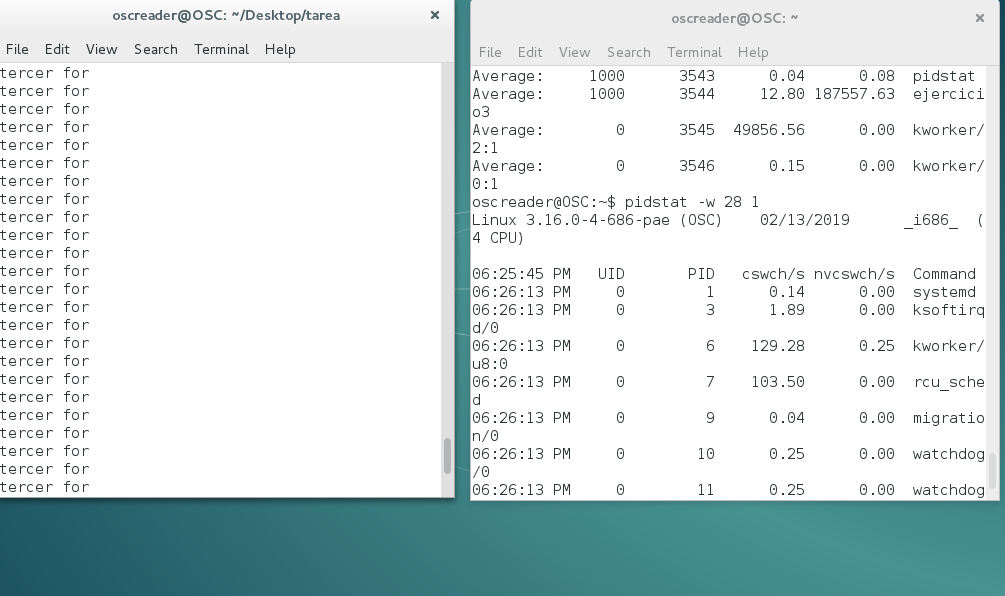
* **¿Qué tipo de cambios de contexto incrementa notablemente en cada caso, y por qué?**

Gnome-shell al abrir una nueva pantalla.

Xorg puede ser afectado por sus dispositivos de entrada como la aceleración del ratón, botones de entrada o teclados.



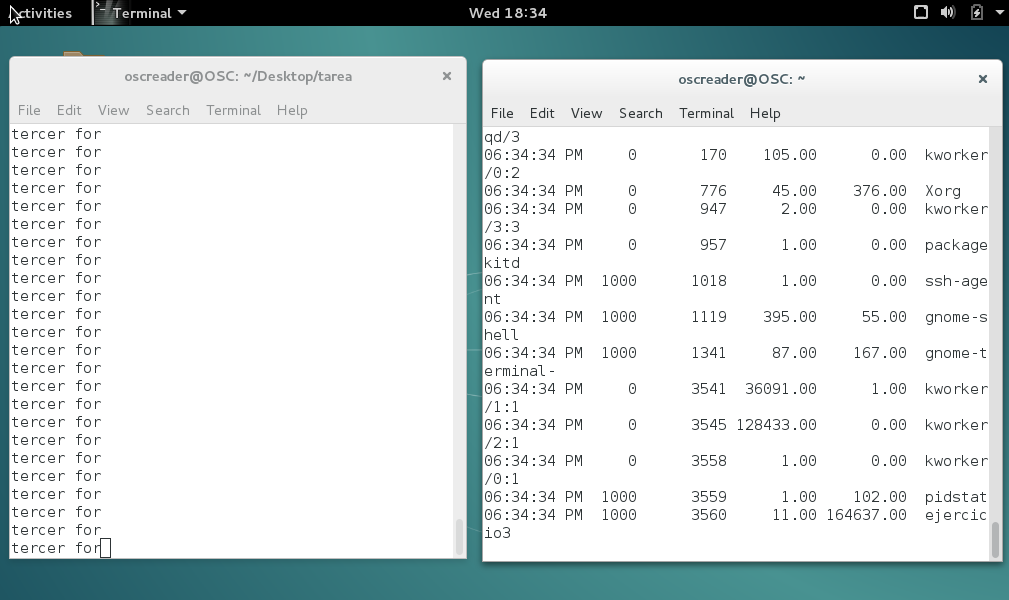




* **¿Qué diferencia hay en el número y tipo de cambios de contexto de entre programas?**

Uno esta mostrando una mayor cantidad de cambios que el otro. Esto se puede deber al hecho que uno lo esta dividiendo en procesos de padres e hijos mientras que el otro no. Otros de los cambios a resaltar es que uno tiene muchos mas datos que el otro, y como tuvo una mayor cantidad de segundos que el otro, tardo mas tiempo en ejecutar.

* **¿A qué puede atribuir los cambios de contexto voluntarios realizados por sus programas?**
* **¿A qué puede atribuir los cambios de contexto involuntarios realizados por sus programas?**
* **¿Por qué el reporte de cambios de contexto para su programa con fork()s muestra cuatro procesos, uno de los cuales reporta cero cambios de contexto?**



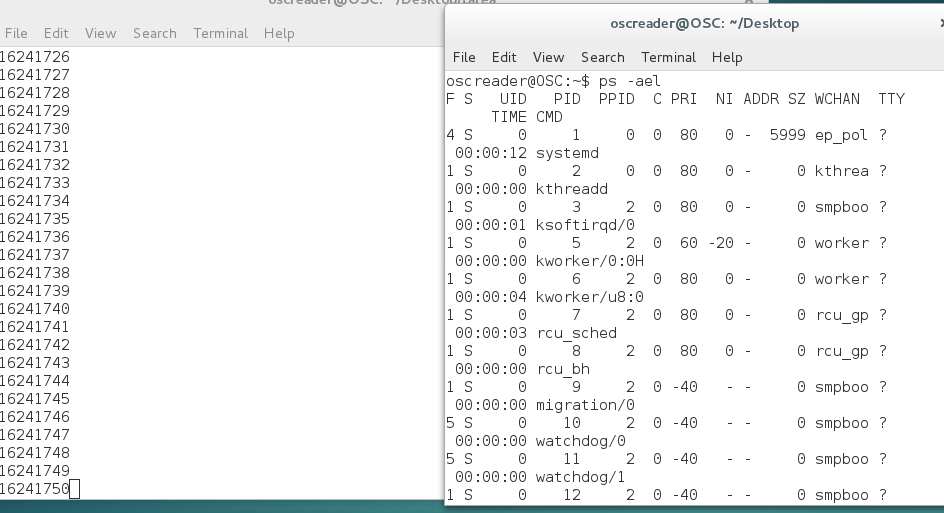
* **¿Qué efecto percibe sobre el número de cambios de contexto de cada tipo?**

Estos nunca terminan y no se mantienen constantes durante su ejecución, y bajan cuando termina la ejecución.

**Ejercicio 4**

* **¿Qué significa la Z y a qué se debe?**

Es un proceso Zombie que termina pero no es cosechado por su padre.



* **¿Qué sucede en la ventana donde ejecutó su programa?**
* **¿Quién es el padre del proceso que quedó huérfano?**

Es adoptado por el init process como un padre.

**Ejercicio 5**

