

**UNIVERSIDAD DE ANTIOQUIA**  
**FACULTAD DE INGENIERÍA**  
**DEPARTAMENTO DE INGENIERÍA DE SISTEMAS**

**SISTEMAS OPERATIVOS Y LABORATORIO**

**PROFESOR JHEISSON ARGIRO LÓPEZ RESTREPO , 2021-2**

**PRIMERA PRACTICA DE LABORATORIO**



Autores:	Daniel Escobar Castañeda	c.c: 1039473937
	Juan Esteban Garcia Muñoz	c.c: 1152717873
	Juan Alejandro Rios Perez	c.c: 1000083984
	Juan Jose Zapata Monsalve	c.c: 1039471932

## 1. Introducción:

Esta práctica tiene como objetivo afianzar los conocimientos vistos en el curso de sistemas operativos en las clases teóricas y de laboratorio. Básicamente consiste en realizar un programa, desarrollado mediante el lenguaje de programación C.

Este, será de utilidad para 3 tareas, la más fundamental, es ver los detalles de un proceso que contenga el sistema operativo en el momento. La segunda parte, se basa en obtener la información de varios procesos al mismo tiempo. Finalmente, la última funcionalidad es a parte de poder ver la información de diferentes procesos, también escribir dicha información en un archivo de texto plano (.txt).

## 2. Consideraciones:

- Para la elaboración de la práctica se hizo uso de la herramienta Virtual Box para virtualizar el sistema operativo Linux, el cual, se exigió para evitar problemas de compatibilidad al momento de la ejecución del programa.
- Fueron muy útiles todas las buenas prácticas de programación que hemos adquirido durante toda la carrera, y más aún, la lógica de programación, fue fundamental para el desarrollo del programa.
- Como lo indicó la etapa 4 de la guía de la práctica. Se hicieron validaciones del programa, para que este, no fallé en casos no esperados como entradas de datos erróneos, falta de agregar datos, etc.

## 3. Preguntas de la guía:

Como ejercicio de práctica, fueron asignadas preguntas teóricas en la guía de la práctica, la solución de estas actividades, están a continuación:

### 1. Abra una terminal de linux y ejecute el comando ps. ¿Qué observa?

```
daniel@daniel-VirtualBox:~$ ps
  PID TTY          TIME CMD
 18280 pts/0    00:00:00 bash
 18788 pts/0    00:00:00 ps
```

**R://** Se observan dos procesos, cuyos nombres son bash y ps. Además, se ve cierta información de los mismos, por ejemplo PID que es el identificador único de cada proceso, TIME que hace referencia al tiempo de ejecución, etc.

### 2. Ejecuta el comando jobs. ¿Qué observa y por qué?

**R://** No se observa nada dado a que con este comando se listan los procesos y como no hay procesos activos, no hay nada que mostrar.

### 3. Ahora llame al editor de texto de su preferencia (en este ejemplo se llamó gedit) en background. Si usted observa la salida verá un número ¿Con qué se relaciona este?

```
daniel@daniel-VirtualBox:~$ gedit&
[1] 18793
```

**R://** Este número se relaciona con el PID (identificador único) del proceso gedit &.

### 4. Ejecute nuevamente el comando jobs. ¿Cambia la salida?

```
daniel@daniel-VirtualBox:~$ jobs
[1]+  Running                  gedit &
```

R:// Ahora como si hay un proceso en ejecución se muestra.

5. Ahora ejecute el comando ps con la opción l

```
daniel@daniel-VirtualBox:~$ ps l
F  UID      PID      PPID  PRI  NI     VSZ   RSS  WCHAN  STAT TTY      TIME  COMMAND
D
4   1000     1289      1237   20   0  170896  6236 do_pol Ssl+  tty2      0:00  /usr/l
0   1000     1294      1289   20   0  230144 14312 do_pol Sl+   tty2      0:00  /usr/l
0   1000    18280     18184   20   0   20260   6012 do_wai Ss    pts/0     0:00  bash
0   1000    18850     18280   20   0  575104 56972 do_pol Sl    pts/0     0:00  gedit
4   1000    19006     18280   20   0   21160   1544 -      R+    pts/0     0:00  ps l
```

¿Qué significado tiene dentro de lo que se ha visto en la teoría cada una de estas salidas? Llene una tabla con la información con estos datos y que le fue arrojada en su caso particular.

R://

- **PID:** Es el identificador único del proceso.
- **VSZ:** Es la cantidad de memoria virtual a la que el proceso tiene acceso.
- **RSS:** Es la cantidad de memoria que se le asigna al proceso
- **size:** Hace relación al tamaño de la sección y de los archivos
- **TTY:** Es el nombre del archivo que su esclavo está usando para interactuar con el maestro

**Status:** Hace referencia al estado del servicio, ya sea que esté disponible o no.

PID	VSZ	RSS	size	TTY	Status
18280	20260	6012		pts/0	
18850	575104	56972		pts/0	
19006	21160	1544		pts/0	

4. Sobre el sistema de archivos procs:

Actividad 1

Actividad 2

```
daniel@daniel-VirtualBox:~$ gedit&
[1] 16944
daniel@daniel-VirtualBox:~$ ls /proc
1      1415   1786   2023   260    2908   590    asound      mdstat
10     1423   1794   2028   2601   2919   599    bootconfig  meminfo
109    1433   18      2029   2602   2969   6      buddyinfo   misc
11     1435   183    2032   2603   297    630    bus         modules
```

**Preguntas:**

1. ¿Qué significan todos estos números enteros? (Consulte el enlace de arriba).

Estos números enteros son los identificadores únicos de cada proceso en ejecución.

2. ¿Estos números son archivos o directorios?

Estos números pertenecen al sistema de archivos proc que es un pseudo-sistema de archivos.

3. El número entero 1 ¿con qué está asociado?

Este número significa que los usuarios no pueden acceder a archivos o subdirectorios dentro de ningún directorio de tipo / proc / [pid], solo tienen acceso a sus propios directorios.

### Actividad 3:

Ahora explore el contenido del número asociado al gedit.

```
daniel@daniel-VirtualBox:~$ ls /proc/16944
arch_status      cwd              mem              patch_state      stat
attr             environ         mountinfo        personality       statm
autogroup        exe             mounts           projid_map       status
auxv             fd              mountstats       root              syscall
cgroup          fdinfo          net              sched             task
clear_refs       gid_map         ns               schedstat         timers_offsets
cmdline          io              numa_maps        sessionid         timers
comm             limits          oom_adj          setgroups         timerslack_ns
coredump_filter  loginuid        oom_score        smaps             uid_map
cpu_resctrl_groups map_files        oom_score_adj    smaps_rollup     wchan
cpuset           maps            pagemap          stack
```

### Preguntas:

1. ¿Qué información se obtiene de los archivos stat y status anteriormente resaltados?

Stat contiene la información del estado del proceso. Sin embargo, Status contiene la información del estado del proceso en lenguaje natural, entendible por nosotros.

2. ¿La información en dichos archivos es la misma?, ¿De cuál archivo es mejor leerla y por qué?

Efectivamente es la misma información en ambos archivos, además es mejor leer esta información del archivo Status ya que muestra la información de manera legible para los humanos.

3. Teniendo en cuenta lo anterior llene la siguiente tabla para el proceso asociado al editor de textos. Llene la siguiente tabla para su caso particular.

Información	Valor
Name	gedit

State	S (sleeping)
FDSize	256
VmPeak	611108 kB
VmSize	575656 kB
VmLck	0 kB
voluntary_ctxt_switches	123
nonvoluntary_ctxt_switches	441

NOTA:

- **Para no extender mucho el informe, sólo incluimos la información referente a las preguntas teóricas expuestas en la guía.**
- **El código está debidamente documentado, siempre hemos usado el idioma inglés para programar, esto con el fin de repasar dicho lenguaje.**
- **En la sustentación, se explicarán las dudas que se tengan respecto al código y su funcionamiento.**