

# Informe de Laboratorio 1

## 1. Análisis de estados de procesos

En una terminal se ejecuta proceso\_estados.py mientras en la otra se observa htop mostrando los procesos del SO

The screenshot shows a desktop environment with a terminal window displaying the output of the `htop` command. The terminal output includes system statistics and a list of running processes. The process `python3 proceso_estados.py` is highlighted in red, showing it is in a 'Running' state with a CPU usage of 132.0%.

```
0[ | 2.6% Tasks: 123, 616 thr, 92 kthr; 2 running
1[ | 0.0% Load average: 0.28 0.14 0.17
2[ | 100.0% Uptime: 00:53:35
3[ | 0.7%
Mem[ | 1.63G/3.82G
Swp[ | 708K/3.82G
```

PID	USER	PRI	NI	VIRT	RES	SHR	S	CPU%	MEM%	TIME+	Command
21426	andres	20	0	29132	9504	6560	R	132.0	0.2	0:10.72	python3 proceso_estados.py
20873	andres	20	0	20692	5612	3820	R	8.1	0.1	1:21.85	htop
2655	andres	20	0	5005M	423M	146M	S	1.6	10.8	1:04.97	/usr/bin/gnome-shell
2728	andres	20	0	5005M	423M	146M	S	0.8	10.8	0:21.26	/usr/bin/gnome-shell
2729	andres	20	0	5005M	423M	146M	S	0.8	10.8	0:21.44	/usr/bin/gnome-shell
20833	andres	20	0	559M	61952	45876	S	0.8	1.5	0:11.04	/usr/libexec/gnome-terminal-server
1	root	20	0	23132	13896	9288	S	0.0	0.3	0:06.98	/sbin/init splash
267	root	19	-1	51240	17884	16476	S	0.0	0.4	0:00.59	/usr/lib/systemd/systemd-journald
344	root	20	0	30544	8300	4844	S	0.0	0.2	0:00.31	/usr/lib/systemd/systemd-udev
381	systemd-oo	20	0	17556	7488	6720	S	0.0	0.2	0:00.77	/usr/lib/systemd/systemd-oomd
382	systemd-re	20	0	21840	13284	10852	S	0.0	0.3	0:00.26	/usr/lib/systemd/systemd-resolved
383	systemd-tl	20	0	91044	7876	6980	S	0.0	0.2	0:00.07	/usr/lib/systemd/systemd-timesyncd
544	systemd-tl	20	0	91044	7876	6980	S	0.0	0.2	0:00.00	/usr/lib/systemd/systemd-timesyncd
918	avahi	20	0	8668	4116	3988	S	0.0	0.1	0:00.15	avahi-daemon: running [andres-VirtualBo
919	messagebus	20	0	12212	7116	4428	S	0.0	0.2	0:02.14	@dbus-daemon --system --address=systemd

**Proceso en ejecución**

Se puede observar el proceso en estado 'Running' (R), indicado por `htop`. Durante esta fase, el proceso utiliza intensivamente la CPU para realizar cálculos, como se ve en la columna CPU%.

The screenshot shows a desktop environment with a terminal window displaying the output of the `htop` command. The terminal output includes system statistics and a list of running processes. The process `python3 proceso_estados.py` is highlighted in red, showing it is in a 'Running' state with a CPU usage of 0.2%.

```
0[ | 1.4% Tasks: 122, 616 thr, 94 kthr; 3 running
1[ | 2.7% Load average: 0.50 0.24 0.20
2[ | 2.1% Uptime: 01:02:40
3[ | 3.5%
Mem[ | 1.62G/3.82G
Swp[ | 708K/3.82G
```

PID	USER	PRI	NI	VIRT	RES	SHR	S	CPU%	MEM%	TIME+	Command
21011	andres	20	0	2256M	239M	115M	S	0.0	6.1	0:00.00	/usr/bin/gnome-text-editor /home/andres
21012	andres	39	19	2256M	239M	115M	S	0.0	6.1	0:00.00	/usr/bin/gnome-text-editor /home/andres
21536	andres	20	0	2881M	62304	47804	S	0.0	1.6	0:00.27	gjs /usr/share/gnome-shell/extensions/d
21538	andres	20	0	2881M	62304	47804	S	0.0	1.6	0:00.00	gjs /usr/share/gnome-shell/extensions/d
21539	andres	20	0	2881M	62304	47804	S	0.0	1.6	0:00.00	gjs /usr/share/gnome-shell/extensions/d
21540	andres	20	0	2881M	62304	47804	S	0.0	1.6	0:00.01	gjs /usr/share/gnome-shell/extensions/d
21541	andres	20	0	2881M	62304	47804	S	0.0	1.6	0:00.00	gjs /usr/share/gnome-shell/extensions/d
21542	andres	20	0	2881M	62304	47804	S	0.0	1.6	0:00.00	gjs /usr/share/gnome-shell/extensions/d
21543	andres	20	0	2881M	62304	47804	S	0.0	1.6	0:00.00	gjs /usr/share/gnome-shell/extensions/d
21544	andres	20	0	2881M	62304	47804	S	0.0	1.6	0:00.00	gjs /usr/share/gnome-shell/extensions/d
21553	andres	20	0	2881M	62304	47804	S	0.0	1.6	0:00.00	gjs /usr/share/gnome-shell/extensions/d
21563	andres	20	0	2881M	62304	47804	S	0.0	1.6	0:00.00	gjs /usr/share/gnome-shell/extensions/d
21575	andres	20	0	29132	9412	6596	S	0.0	0.2	0:58.94	python3 proceso_estados.py
21580	andres	20	0	11.3G	459M	210M	S	0.0	11.7	0:00.06	/snap/firefox/5751/usr/lib/firefox/fire
							S	0.0	2.3	0:00.00	/snap/firefox/5751/usr/lib/firefox/fire

**El proceso se suspende/bloquea**

El mismo proceso ahora tendrá una **s** en la columna de estado, que significa "Sleeping" (Interrumpible). El uso de CPU bajará a 0% porque está esperando, no calculando.

**El proceso finaliza (ya no se muestra en el terminal)**

```
.)
Estado: Terminado
andres@andres-VirtualBox:~/Escritorio$ python3 proceso_estados.py
Inicio del script. PID: 21575
```

Terminal

```
0[ ||| 3.4% Tasks: 123, 621 thr, 96 kthr; 1 running
1[ 0.6% Load average: 0.39 0.25 0.20
2[ 0.7% Uptime: 01:05:49
3[ 0.7%
Mem[ ||||| 1.65G/3.82G
Swp[ | 708K/3.82G
```

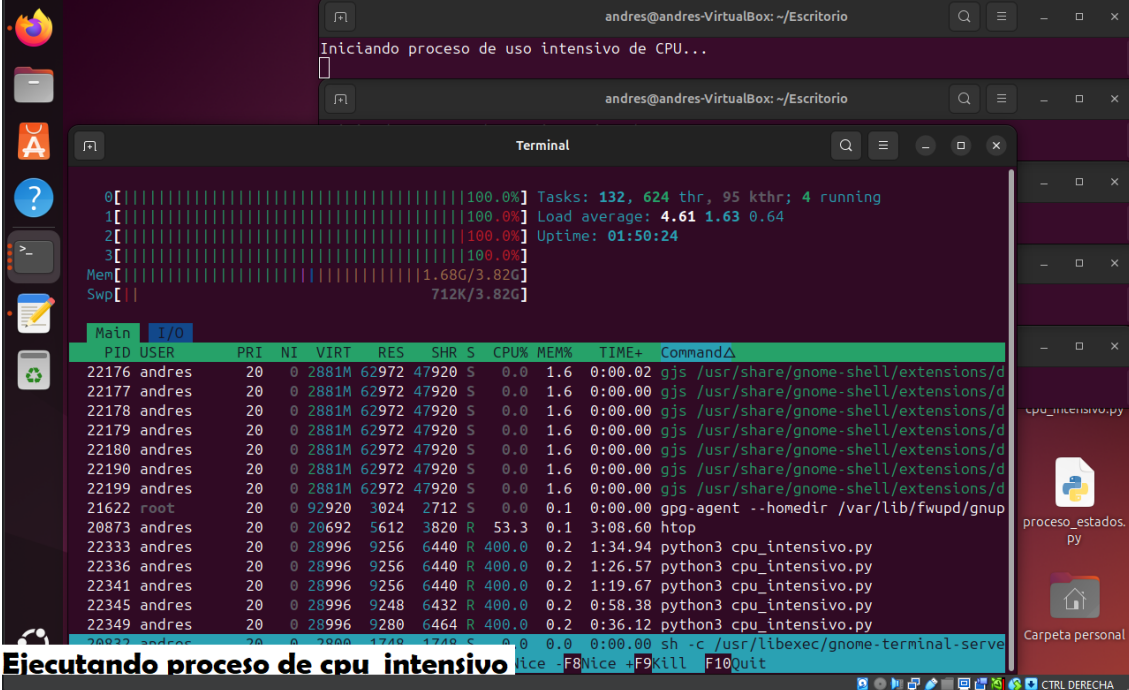
Main		I/O																	
PID	USER	PRI	NI	VIRT	RES	SHR	S	CPU%	MEM%	TIME+	Command								
2831	andres	20	0	388M	12464	7260	S	0.0	0.3	0:01.11	/usr/bin/ibus-daemon --panel disable								
3041	andres	20	0	388M	12464	7260	S	0.0	0.3	0:00.00	/usr/bin/ibus-daemon --panel disable								
3042	andres	20	0	388M	12464	7260	S	0.0	0.3	0:00.00	/usr/bin/ibus-daemon --panel disable								
3046	andres	20	0	388M	12464	7260	S	0.0	0.3	0:01.93	/usr/bin/ibus-daemon --panel disable								
2371	andres	20	0	121M	14292	8660	S	0.0	0.4	0:00.13	/usr/bin/pipewire								
2399	andres	20	0	121M	14292	8660	S	0.0	0.4	0:00.00	/usr/bin/pipewire								
2412	andres	20	0	121M	14292	8660	S	0.0	0.4	0:00.21	/usr/bin/pipewire								
2372	andres	20	0	104M	5832	5064	S	0.0	0.1	0:00.00	/usr/bin/pipewire -c filter-chain.conf								
2400	andres	20	0	104M	5832	5064	S	0.0	0.1	0:00.00	/usr/bin/pipewire -c filter-chain.conf								
2404	andres	20	0	104M	5832	5064	S	0.0	0.1	0:00.00	/usr/bin/pipewire -c filter-chain.conf								
2377	andres	20	0	125M	16720	10832	S	0.0	0.4	0:00.10	/usr/bin/pipewire-pulse								
2397	andres	20	0	125M	16720	10832	S	0.0	0.4	0:00.00	/usr/bin/pipewire-pulse								
2402	andres	20	0	125M	16720	10832	S	0.0	0.4	0:00.02	/usr/bin/pipewire-pulse								
1169	root	20	0	118M	22668	13452	S	0.0	0.6	0:00.06	/usr/bin/python3 /usr/share/unattended-								
1176	root	20	0	118M	22668	13452	S	0.0	0.6	0:00.00	/usr/bin/python3 /usr/share/unattended-								

F3Next S-F3Prev EscCancel Search: python3

Los estados teóricos **Nuevo y Listo** son transiciones muy rápidas que **htop** no siempre puede mostrar, pero el estado Terminado se evidencia cuando el proceso desaparece.

## 2. Análisis de Scheduling del SO.

Se ejecutaron 5 procesos que demandan el 100% de la CPU para observar la respuesta del planificador.



```
andres@andres-VirtualBox: ~/Escritorio
Iniciando proceso de uso intensivo de CPU...

0[|||||100.0%] Tasks: 132, 624 thr, 95 kthr; 4 running
1[|||||100.0%] Load average: 4.61 1.63 0.64
2[|||||100.0%] Uptime: 01:50:24
3[|||||100.0%]
Mem[|||||1.68G/3.82G]
Swp[|||||712K/3.82G]

Main I/O
PID USER PRI NI VIRT RES SHR S CPU% MEM% TIME+ Command
22176 andres 20 0 2881M 62972 47920 S 0.0 1.6 0:00.02 gjs /usr/share/gnome-shell/extensions/d
22177 andres 20 0 2881M 62972 47920 S 0.0 1.6 0:00.00 gjs /usr/share/gnome-shell/extensions/d
22178 andres 20 0 2881M 62972 47920 S 0.0 1.6 0:00.00 gjs /usr/share/gnome-shell/extensions/d
22179 andres 20 0 2881M 62972 47920 S 0.0 1.6 0:00.00 gjs /usr/share/gnome-shell/extensions/d
22180 andres 20 0 2881M 62972 47920 S 0.0 1.6 0:00.00 gjs /usr/share/gnome-shell/extensions/d
22190 andres 20 0 2881M 62972 47920 S 0.0 1.6 0:00.00 gjs /usr/share/gnome-shell/extensions/d
22199 andres 20 0 2881M 62972 47920 S 0.0 1.6 0:00.00 gjs /usr/share/gnome-shell/extensions/d
21622 root 20 0 92920 3024 2712 S 0.0 0.1 0:00.00 gpg-agent --homedir /var/lib/fwupd/gnup
20873 andres 20 0 20692 5612 3820 R 53.3 0.1 3:08.60 htop
22333 andres 20 0 28996 9256 6440 R 400.0 0.2 1:34.94 python3 cpu_intensivo.py
22336 andres 20 0 28996 9256 6440 R 400.0 0.2 1:26.57 python3 cpu_intensivo.py
22341 andres 20 0 28996 9256 6440 R 400.0 0.2 1:19.67 python3 cpu_intensivo.py
22345 andres 20 0 28996 9248 6432 R 400.0 0.2 0:58.38 python3 cpu_intensivo.py
22349 andres 20 0 28996 9280 6464 R 400.0 0.2 0:36.12 python3 cpu_intensivo.py
20039 andres 20 0 20000 1740 1740 C 0.0 0.0 0:00.00 sh -c /usr/libexec/gnome-terminal-serve
Nice +F8 Nice +F9 Kill F10 Quit
```

Como se ve en la captura, el Sistema Operativo Linux no permite que un solo proceso monopolice la CPU. En su lugar, distribuye el tiempo de procesamiento entre los cinco procesos de manera equitativa. Esto es consistente con los algoritmos de planificación modernos como Round Robin o, más específicamente en Linux, el CFS, que busca dar a cada proceso una porción 'justa' de la CPU.

### 3. Simulación y análisis de deadlock

El deadlock sería dos o más procesos esperando por un recurso que tiene el otro.

El script simula esto, el Proceso 1 toma Lock A y espera por B, mientras Proceso 2 toma Lock B y espera por A.

The screenshot shows a Linux desktop with a terminal window titled 'andres@andres-VirtualBox: ~/Escritorio'. The terminal displays the output of a Python script named 'deadlock\_script.py'. The script simulates two processes: Proceso 1 and Proceso 2. Proceso 1 acquires lock\_a and then waits for lock\_b. Proceso 2 acquires lock\_b and then waits for lock\_a. The script output shows the processes waiting for each other, resulting in a deadlock.

```
andres@andres-VirtualBox:~/Escritorio$ python3 deadlock_script.py
Proceso 1: Intentando adquirir lock_a...
Proceso 1: Adquirió lock_a. Esperando 1 seg...
Proceso 2: Intentando adquirir lock_b...
Esperando que los hilos terminen...
Proceso 2: Adquirió lock_b. Esperando 1 seg...
Proceso 2: Intentando adquirir lock_a...
Proceso 1: Intentando adquirir lock_b...
```

Below the terminal window, a process list is displayed, showing the state of the system. The process list includes columns for PID, USER, PRI, NI, VIRT, RES, SHR, S, CPU%, MEM%, TIME+, and Command. The processes are listed in a table format.

PID	USER	PRI	NI	VIRT	RES	SHR	S	CPU%	MEM%	TIME+	Command
22448	andres	20	0	2881M	62804	47888	S	0.0	1.6	0:00.00	gjs /usr/share/gnome-shell/extensions/d
22449	andres	20	0	2881M	62804	47888	S	0.0	1.6	0:00.00	gjs /usr/share/gnome-shell/extensions/d
22450	andres	20	0	2881M	62804	47888	S	0.0	1.6	0:00.01	gjs /usr/share/gnome-shell/extensions/d
22451	andres	20	0	2881M	62804	47888	S	0.0	1.6	0:00.00	gjs /usr/share/gnome-shell/extensions/d
22453	andres	20	0	2881M	62804	47888	S	0.0	1.6	0:00.00	gjs /usr/share/gnome-shell/extensions/d
22454	andres	20	0	2881M	62804	47888	S	0.0	1.6	0:00.00	gjs /usr/share/gnome-shell/extensions/d
22455	andres	20	0	2881M	62804	47888	S	0.0	1.6	0:00.00	gjs /usr/share/gnome-shell/extensions/d
22465	andres	20	0	2881M	62804	47888	S	0.0	1.6	0:00.00	gjs /usr/share/gnome-shell/extensions/d
22474	andres	20	0	2881M	62804	47888	S	0.0	1.6	0:00.00	gjs /usr/share/gnome-shell/extensions/d
21622	root	20	0	92920	3024	2712	S	0.0	0.1	0:00.00	gpg-agent --homedir /var/lib/fwupd/gnup
20873	andres	20	0	20692	5612	3820	R	0.7	0.1	3:27.05	htop
22533	andres	20	0	172M	10476	6764	S	0.0	0.3	0:00.01	python3 deadlock_script.py
22534	andres	20	0	172M	10476	6764	S	0.0	0.3	0:00.00	python3 deadlock_script.py
22535	andres	20	0	172M	10476	6764	S	0.0	0.3	0:00.00	python3 deadlock_script.py
20832	andres	20	0	2800	1748	1748	S	0.0	0.0	0:00.00	sh -c /usr/libexec/gnome-terminal-serve

The process list shows that the processes are in a state of deadlock, with the CPU usage for the processes being 0.0% and the memory usage being 0.3%.

**El proceso se quedo colgado**

La captura muestra que el programa se detiene indefinidamente, demostrando una condición de deadlock. El SO no interviene para resolver este tipo de deadlock a nivel de aplicación; la responsabilidad recae en el programador. La única forma de resolverlo fue terminar los procesos manualmente (Ctrl+C), liberando así los recursos.