

Práctica II



Alfonso García Jorge

1ºDAM

Índice

Paquetes, Repositorios e Instalación de software en GNU/Linux.....	4
Ejercicio A.1.....	4
Ejercicio A.2.....	8
Ejercicio A.3.....	10
Procesos.....	11
Ejercicio B.1.....	11
Ejercicio B.2.....	14
Ejercicio B.3.....	19
Servicios.....	20
Ejercicio C.1.....	20
Ejercicio C.2.....	22
Ejercicio C.3.....	23

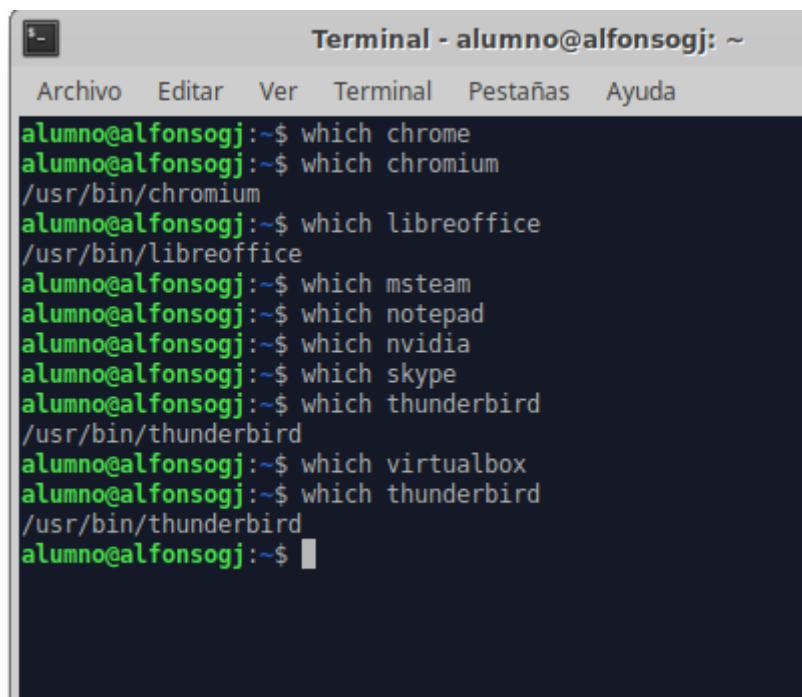
Paquetes, Repositorios e Instalación de software en GNU/Linux

Ejercicio A.1

Instala el siguiente software en la máquina virtual de GNU/Linux de clase. Para cada programa responde a las siguientes preguntas e indica qué comandos usaste y toma capturas de pantalla que muestren el resultado:

1. ¿Cómo compruebas si el programa estaba instalado o no? Si estaba instalado, comprueba si lo puedes actualizar. Si no estaba instalado, instálalo y muestra que funciona correctamente.

`dpkg -l | grep paquete` ó `which paquete`

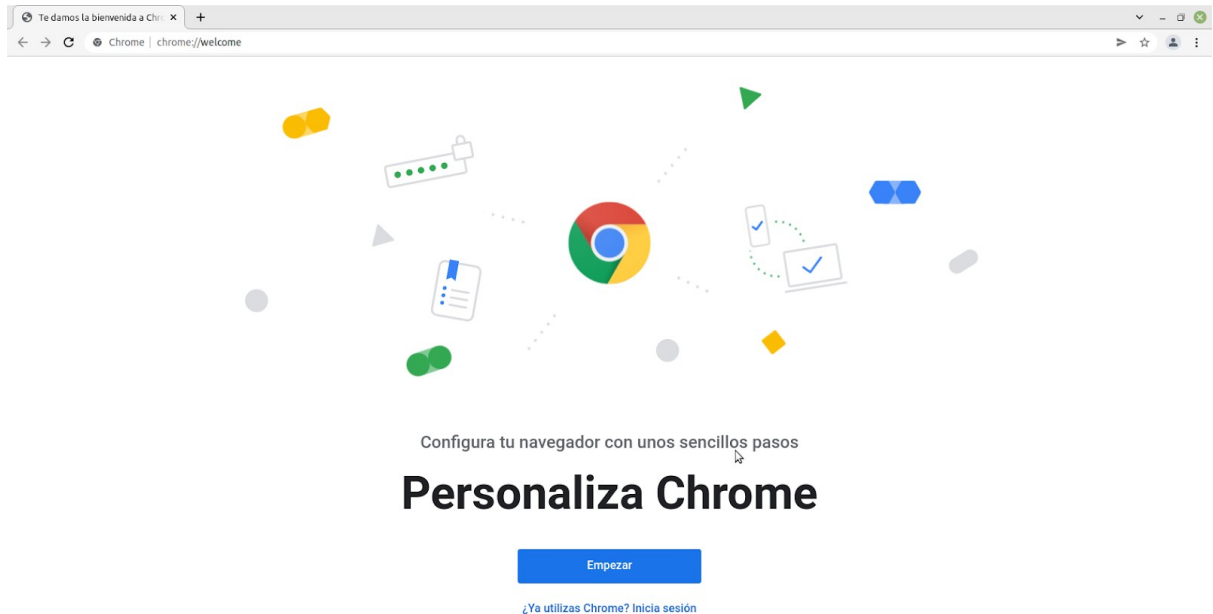


```
Terminal - alumno@alfonsogj: ~
Archivo  Editar  Ver  Terminal  Pestañas  Ayuda
alumno@alfonsogj:~$ which chrome
alumno@alfonsogj:~$ which chromium
/usr/bin/chromium
alumno@alfonsogj:~$ which libreoffice
/usr/bin/libreoffice
alumno@alfonsogj:~$ which msteam
alumno@alfonsogj:~$ which notepad
alumno@alfonsogj:~$ which nvidia
alumno@alfonsogj:~$ which skype
alumno@alfonsogj:~$ which thunderbird
/usr/bin/thunderbird
alumno@alfonsogj:~$ which virtualbox
alumno@alfonsogj:~$ which thunderbird
/usr/bin/thunderbird
alumno@alfonsogj:~$
```

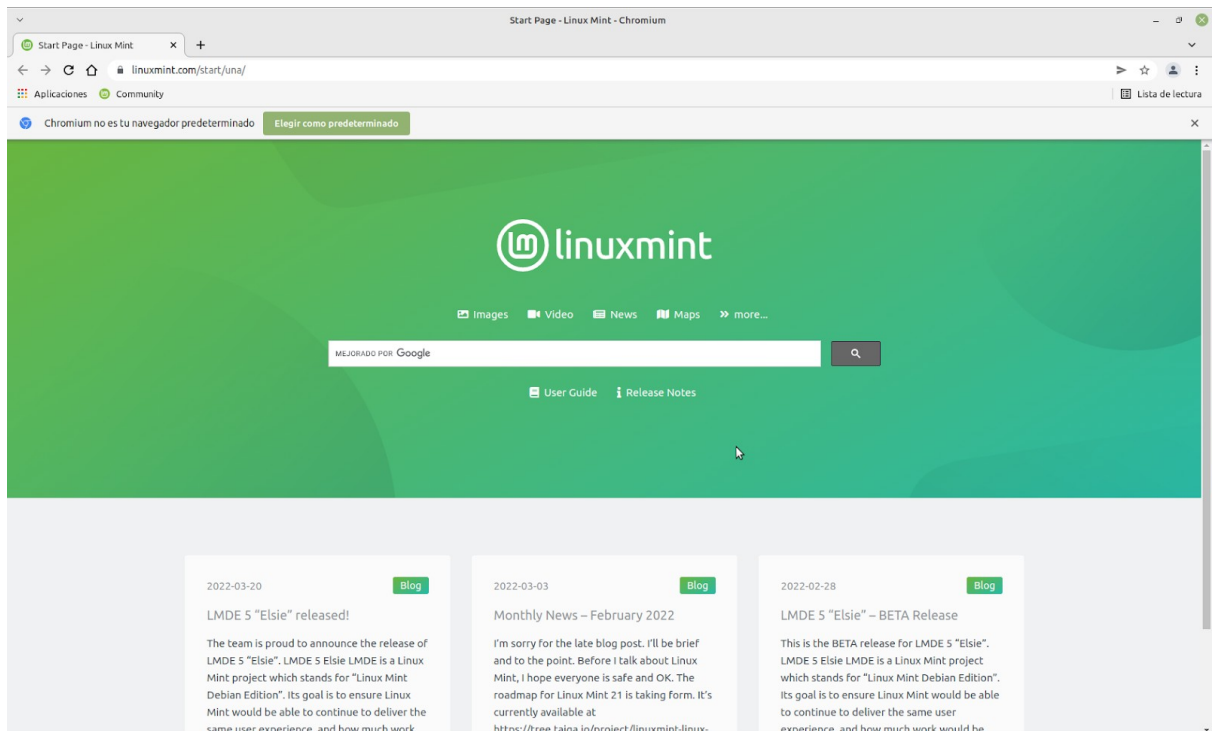
2. ¿Qué versión estaba instalada (si había alguna) y cuál es la que se va a instalar? ¿Has instalado la última versión disponible del software o una más antigua (si puedes, indica la versión instalada y la más reciente, indicando incluso la diferencia de fechas si conoces ese dato)?
3. ¿Cómo instalaste el programa? (repositorio, paquete descargado, PPA, etc.) ¿por qué elegiste esta opción?
4. ¿Estaba el programa en algún repositorio? Si es afirmativo, ¿en qué repositorio? ¿por qué estaba justo en ese repositorio y no en otro?
5. ¿Sabes qué dependencias, recomendaciones y/o sugerencias tiene el software a actualizar?
6. Elimina totalmente el nuevo software instalado SI NO LO TENÍAS PREVIAMENTE

SOFTWARE A INSTALAR:

1. Chrome

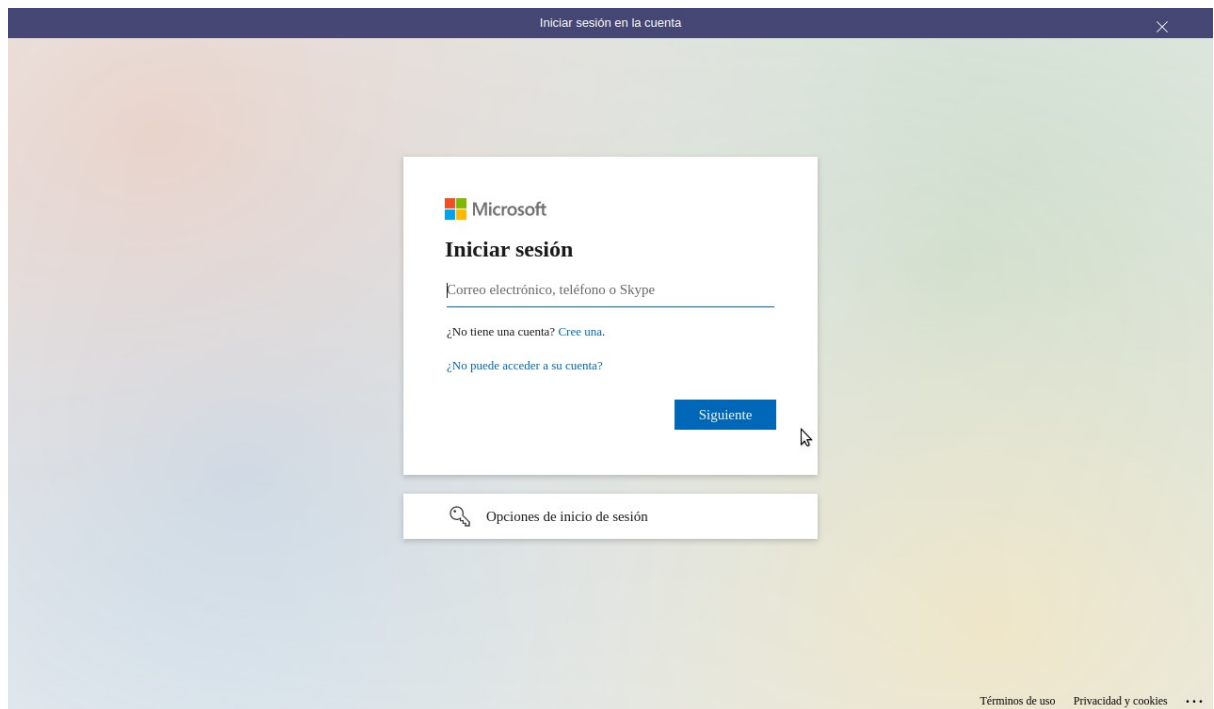


2. Chromium



3. LibreOffice 7.3.x

4. MS Teams



5. notepad++ (o equivalente)

6. nvidia-utils

7. Skype

8. Thunderbird

9. Un antivirus

10. Virtualbox

PASO FINAL (al acabar el ejercicio):

1. Cuando acabes el ejercicio ¿qué paquetes quedan en el sistema que ya no son necesarios? Elimina todos estos paquetes innecesarios
2. Revisa la lista de repositorios que tienes activos en el sistema. Elimina o desactiva los repositorios PPA o de terceros del software que has instalado y luego desinstalado en este ejercicio.

Se realiza con el comando `apt-get -purge remove` “”

1. Chrome

Se debe descargar de forma manual. Al estar descargado para abrirlo desde terminal ponemos google-chrome.

2. Chromium

Se instala desde repositorio, ya que es oficial por GNU. Recomiendan las dependencias para desarrolladores.

```
alumno@alfonsogj:~$ sudo apt-get install chromium
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
chromium ya está en su versión más reciente (90.0.4430.212-1~deb10u1).
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 11 no actualizados.
alumno@alfonsogj:~$
```

```
alumno@alfonsogj:~$ chromium --version
Chromium 90.0.4430.212 built on Debian 10.9, running on Debian bullseye/sid
alumno@alfonsogj:~$
```

3. LibreOffice

Se instala desde repositorio, oficial por GNU.

```
alumno@alfonsogj:~$ sudo apt-get install libreoffice
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  firebird3.0-common firebird3.0-common-doc firebird3.0-server-core
  firebird3.0-utils fonts-crosextra-caladea fonts-crosextra-carlito
alumno@alfonsogj:~$
```

```
alumno@alfonsogj:~$ libreoffice
alumno@alfonsogj:~$ libreoffice --version
LibreOffice 6.4.7.2 40(Build:2)
alumno@alfonsogj:~$
```

4. MS Teams

Con el comando `sudo apt -f install ./teams_1.3.00.16851_amd64.deb -y`

5. Notepad++

Se descarga el snap.

```
alumno@alfonsogj:~$ snap install notepad-plus-plus
Montar snap "notepad-plus-plus" (356)
```

6. Nvidia-utils

7.Skype

Se descarga el paquete: `wget https://go.skype.com/skypeforlinux-64.deb`

```
alumno@alfonsogj:~$ wget https://go.skype.com/skypeforlinux-64.deb
--2022-03-24 00:08:36-- https://go.skype.com/skypeforlinux-64.deb
Resolviendo go.skype.com (go.skype.com)... 40.118.109.53
Conectando con go.skype.com (go.skype.com)[40.118.109.53]:443... conectado.
Petición HTTP enviada, esperando respuesta... 301 Moved Permanently
Ubicación: https://get.skype.com/getskype-webwrap-deb [siguiente]
--2022-03-24 00:08:37-- https://get.skype.com/getskype-webwrap-deb
Resolviendo get.skype.com (get.skype.com)... 52.174.193.75
Conectando con get.skype.com (get.skype.com)[52.174.193.75]:443... conectado.
Petición HTTP enviada, esperando respuesta... 302 Found
Ubicación: https://repo.skype.com/latest/skypeforlinux-64.deb [siguiente]
--2022-03-24 00:08:37-- https://repo.skype.com/latest/skypeforlinux-64.deb
Resolviendo repo.skype.com (repo.skype.com)... 184.24.68.146, 2a02:26f0:9f00:6
::1263, 2a02:26f0:9f00:6a5::1263
Conectando con repo.skype.com (repo.skype.com)[184.24.68.146]:443... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: 140790360 (134M) [application/x-debian-package]
Guardando como: "skypeforlinux-64.deb"
```

```
alumno@alfonsogj:~$ sudo dpkg -i skypeforlinux-64.deb
Seleccionando el paquete skypeforlinux previamente no seleccionado.
(Leyendo la base de datos ... 256890 ficheros o directorios instalados actualmen
te.)
Preparando para desempaquetar skypeforlinux-64.deb ...
Desempaquetando skypeforlinux (8.82.0.403) ...
Configurando skypeforlinux (8.82.0.403) ...
Procesando disparadores para gnome-menus (3.36.0-lubuntu1) ...
Procesando disparadores para desktop-file-utils (0.24-lubuntu3) ...
Procesando disparadores para mime-support (3.64ubuntu1) ...
Procesando disparadores para hicolor-icon-theme (0.17-2) ...
alumno@alfonsogj:~$
```

8.Thunderbird

Debemos descargar thunderbird desde la página en tu directorio de inicio.

Luego debemos descomprimirlo: `tar xjf thunderbird-*.tar.bz2`

Para iniciar Thunderbird, ejecuta el script thunderbird en la carpeta thunderbird: `~/thunderbird/thunderbird`

9.Antivirus: Sophos

Descargar desde su página.

Descomprimir con `tar -xvzf fav-linux-free-9.tgz`

Nos metemos dentro del subdirectorio `cd sophos-av`

Ejecutamos el script: `sudo ./install.sh`

Para iniciar la protección: `sudo etcinit/sav-protect start`

9.VirtualBox

Desde repositorio oficiales

```
Procesando disparadores para nicotol-icon-theme (0.17~2) ...
alumno@alfonsogj:~$ sudo apt-get install virtualbox
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
dctrl-tools dkms libdouble-conversion3 libgsoap-2.8.91 liblzfl1 libpcre2-16-0
libqt5core5a libqt5dbus5 libqt5gui5 libqt5network5 libqt5opengl5
libqt5printsupport5 libqt5svg5 libqt5widgets5 libqt5xmlextras5
libstdl1.2debian libxcb-xinerama0 libxcb-xinput0 qt5-gtk-platformtheme
qttranslations5-l10n virtualbox-dkms virtualbox-qt
Paquetes sugeridos:
debtags menu qt5-image-formats-plugins qtwayland5 vde2
virtualbox-guest-additions-iso
Se instalarán los siguientes paquetes NUEVOS:
dctrl-tools dkms libdouble-conversion3 libgsoap-2.8.91 liblzfl1 libpcre2-16-0
libqt5core5a libqt5dbus5 libqt5gui5 libqt5network5 libqt5opengl5
libqt5printsupport5 libqt5svg5 libqt5widgets5 libqt5xmlextras5
libstdl1.2debian libxcb-xinerama0 libxcb-xinput0 qt5-gtk-platformtheme
qttranslations5-l10n virtualbox virtualbox-dkms virtualbox-qt
0 actualizados, 23 nuevos se instalarán, 0 para eliminar y 26 no actualizados.
Se necesita descargar 55,1 MB de archivos.
Se utilizarán 226 MB de espacio de disco adicional después de esta operación.
```


Ejercicio A.2

Indica en qué directorio(s), en general, encontrarías:

1. *Los binarios y las librerías de las aplicaciones del usuario instaladas desde repositorio*

Los binarios en /usr/bin

Las bibliotecas en /usr/local/lib

2. *Las aplicaciones instaladas de otras fuentes (no en repositorios oficiales)*

En /usr/local/bin

3. *La configuración de las aplicaciones*

En /etc

4. *El manual de las aplicaciones*

En /usr/man

5. *Los logs o registros de las aplicaciones*

En /var/log

6. *Los binarios del sistema (los utilizados por los usuarios y los utilizados por el propio sistema)*

El directorio /sbin contiene los comandos necesarios para bootear el sistema, pero usualmente no son ejecutados por usuarios normales.

7. *Las librerías del sistema*

Se guarda en /lib

8. *Dispositivos externos (pendrives, disco duro externo, etc.)*

Los dispositivos se encuentran dentro del /dev

9. *Los ficheros temporales*

En /tmp

10. *Dispositivos del sistema*

En /dev

11. *El espacio personal de los usuarios*

En /home/usr

12. *El espacio personal del administrador*

Es /root

13. *Datos del sistema (firmware, kernel, etc.)*

Se guardan en /etc/conf.modules

14.Datos de los procesos del sistema

En /proc

15.Datos del arranque del sistema

En /boot

Ejercicio A.3

Crear un script que:

1. *Debe recibir un argumento (el nombre de un paquete). Si no lo recibe, abortará y mostrará la ayuda*
2. *Comprobará si el paquete indicado se encuentra instalado en el sistema. Si es así, indicar la versión instalada.*
3. *Si el paquete está instalado, se mostrará un menú con opciones sobre ese paquete (implementar cada operación): Actualizar, Reinstalar, Eliminar dejando configuración, Eliminar totalmente, Salir.*
4. *Si no está instalado, comprobar que el paquete está disponible para ser instalado desde algún repositorio. Si es así, ofrecerle la opción para instalarlo. Si no está disponible, mostrarle el resultado de una búsqueda del parámetro indicado.*

Procesos

Ejercicio B.1

Resuelve en GNU/Linux los siguientes apartados. Cuando se pregunte por qué comando se utilizaría, debes indicar el comando completo, con las opciones y parámetros necesarios para que funcione:

1. Ejecutar el comando `ps aux` e indicar qué significa cada columna. ¿Qué estados puede tener un proceso y qué significa cada uno? (si se tienen dudas, usar `man ps` o `ps --help`)

El comando `ps` monitoriza todos los procesos.

La primera columna nos muestra los usuarios. El PID el ID del proceso. El %CPU porcentaje de tiempo que el proceso estuvo en ejecución. El %MEM porcentaje de memoria física utilizada. El VSZ memoria virtual del proceso medida en KiB. El RSS es la cantidad de memoria física no swappeada que la tarea a utilizado. EL TT temrinal que controla el proceso. El STAT codigo de estado del proceso. El TIME tiempo de CPU acumulado. El COMMAND comando con todos sus argumentos.

Los códigos de estado del proceso pueden ser:

- D: suspendido no interrumpible (generalmente esperando E/S)
- R: en ejecución o listo para ejecutarse (en cola)
- S: suspendido interrumpible (esperando que se complete un evento)
- T: detenido, por una señal de control de trabajo o porque está siendo traceado
- W: paginado (no válido a partir del kernel 2.6.xx)
- X: muerto (nunca debe ser visto)
- Z: proceso difunto ("zombie"), terminado pero no reclamado por el proceso padre

```
alumno@alfonsogj:~$ ps aux
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.7  0.3 168732 12620 ?        Ss   22:36   0:05 /sbin/init splash
root         2  0.0  0.0      0     0 ?        S    22:36   0:00 [kthreadd]
root         3  0.0  0.0      0     0 ?        I<   22:36   0:00 [rcu_gp]
root         4  0.0  0.0      0     0 ?        I<   22:36   0:00 [rcu_par_gp]
root         6  0.0  0.0      0     0 ?        I<   22:36   0:00 [kworker/0:0H-events_highpri]
root         8  0.0  0.0      0     0 ?        I    22:36   0:00 [kworker/u2:0-ext4-rsv-conversion]
root         9  0.0  0.0      0     0 ?        I<   22:36   0:00 [mm_percpu_wq]
root        10  0.0  0.0      0     0 ?        S    22:36   0:00 [rcu_tasks_rude_]
root        11  0.0  0.0      0     0 ?        S    22:36   0:00 [rcu_tasks_trace]
root        12  0.1  0.0      0     0 ?        S    22:36   0:00 [ksoftirqd/0]
root        13  0.0  0.0      0     0 ?        I    22:36   0:00 [rcu_sched]
root        14  0.0  0.0      0     0 ?        S    22:36   0:00 [migration/0]
root        15  0.0  0.0      0     0 ?        S    22:36   0:00 [idle_inject/0]
root        16  0.0  0.0      0     0 ?        S    22:36   0:00 [cpuhp/0]
root        17  0.0  0.0      0     0 ?        S    22:36   0:00 [kdevtmpfs]
root        18  0.0  0.0      0     0 ?        I<   22:36   0:00 [netns]
root        19  0.0  0.0      0     0 ?        I<   22:36   0:00 [inet_frag_wq]
root        20  0.0  0.0      0     0 ?        S    22:36   0:00 [kauditd]
root        21  0.0  0.0      0     0 ?        S    22:36   0:00 [khungtaskd]
root        22  0.0  0.0      0     0 ?        S    22:36   0:00 [oom_reaper]
root        23  0.0  0.0      0     0 ?        I<   22:36   0:00 [writeback]
root        24  0.0  0.0      0     0 ?        S    22:36   0:00 [kcompactd0]
root        25  0.0  0.0      0     0 ?        SN   22:36   0:00 [ksmd]
root        26  0.0  0.0      0     0 ?        SN   22:36   0:00 [khugepaged]
root        72  0.0  0.0      0     0 ?        I<   22:36   0:00 [kintegrityd]
root        73  0.0  0.0      0     0 ?        I<   22:36   0:00 [kblockd]
root        74  0.0  0.0      0     0 ?        I<   22:36   0:00 [blkcg_punt_bio]
root        75  0.0  0.0      0     0 ?        I<   22:36   0:00 [tpm_dev_wq]
root        76  0.0  0.0      0     0 ?        I<   22:36   0:00 [ata_sff]
root        77  0.0  0.0      0     0 ?        I<   22:36   0:00 [md]
root        78  0.0  0.0      0     0 ?        I<   22:36   0:00 [edac-poller]
root        79  0.0  0.0      0     0 ?        I<   22:36   0:00 [devfreq_wq]
root        80  0.0  0.0      0     0 ?        S    22:36   0:00 [watchdogd]
root        81  0.0  0.0      0     0 ?        I    22:36   0:00 [kworker/u2:1-ext4-rsv-conversion]
root        82  0.0  0.0      0     0 ?        I<   22:36   0:00 [kworker/0:1H-kblockd]
root        84  0.1  0.0      0     0 ?        S    22:36   0:01 [kswapd0]
root        85  0.0  0.0      0     0 ?        S    22:36   0:00 [ecryptfs-kthrea]
root        87  0.0  0.0      0     0 ?        I<   22:36   0:00 [kthrotld]
```

2. Ejecutar el comando `uptime`. ¿Qué significa cada uno de los datos que muestra y qué información puedo obtener de ellos?

```
alumno@alfonsogj:~$ uptime
23:08:39 up 32 min,  1 user,  load average: 0,20, 0,23, 0,27
alumno@alfonsogj:~$
```

El 23:08 nos muestra la hora actual, up 32 min el tiempo que lleva la maquina encendida. 1 User el numero de usuarios conectados y load average es la carga del sistema actual al minutos, cinco minutos y quince minutos.

3. ¿Un sistema con 8 cores tiene una carga de 5.3, a qué porcentaje aproximado de rendimiento está funcionando este equipo?

Sería un 66% aprox.

4. ¿Qué información me muestra el comando `jobs`? ¿Qué significan los símbolos + y - que aparecen en la lista?

El comando `jobs` nos muestra los trabajos o procesos vinculados a la terminal desde donde se ejecuta dicho comando.

En lo referente a los simbolos + y - si observamos el comando `jobs` al realizar un `bg` o un `bg +` este pondrá a ejecutarse al último proceso añadido a la lista en cambio si hacemos `bg -` afectará al penúltimo proceso añadido a la lista (Los + y - van cambiando de posición ya que al realizar un cambio en el proceso - este pasará a ser el último en recibir el + de nuevo).

5. Al ejecutar el comando `jobs` veo que, por ejemplo, el elemento [3] de la lista está en estado *stopped* (Terminado, *terminated* o similar), ¿qué significa esto? ¿cómo puedo hacer que pase a ejecutarse en segundo plano?

Significa que el proceso se encuentra detenido, para poder pasarlo a segundo plano debemos realizar un `ctrl+`

6. He lanzado un comando `xeyes` que ya se está ejecutando en primer plano, indicar qué tendría que hacer para pasarlo a ejecutar en segundo plano (y que siga ejecutándose)

Con `bg xeyes`

7. Lanzar el comando `xeyes` en segundo plano, y luego volver a pasarlo a primer plano.

Con `fg num` para pasarlo a primer plano

8. ¿Cómo puedo hacer que el comando `xeyes` se siga ejecutando aunque se cierre la shell en la que se ejecutó si aún no lo he lanzado?

Lanzando el `xeyes` con el comando `nohup xeyes`.

9. ¿Cómo puedo hacer que el comando `xeyes` se siga ejecutando aunque se cierre la shell en la que se ejecutó si ya se está ejecutando?

Podemos usar el comando `disown xeyes`.

10. Como usuario tienes que lanzar el comando `xeyes` con la mínima prioridad posible, ¿qué comando utilizarías? ¿qué valor le asignarías y cuánta es la prioridad efectiva final? ¿por qué? ¿qué pasaría en el caso inverso, que quisiera asignarle la máxima prioridad posible antes de lanzarlo?

Para otorgarle prioridad mínima usaremos: nice -19 xeyes

Para otorgarle prioridad máxima usaremos: nice -20 xeyes

11. Repetir el ejercicio anterior, pero sobre el comando xeyes que ya se está ejecutando (indicar qué comando hay que utilizar para modificar la prioridad de un proceso que ya está en ejecución, y qué valores máximos y mínimos se pueden asignar)

Si se está ejecutando sería lo mismo pero con renice y el PID del proceso.

Ejercicio B.2

Ejercicio de procesos en MS Windows:

1. Lista todos los procesos actuales de Windows

Usaremos el comando tasklist.

```
Símbolo del sistema
Microsoft Windows [Versión 10.0.22000.556]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\Alfonso>tasklist

Nombre de imagen                PID Nombre de sesión Núm. de ses Uso de memor
=====
System Idle Process             0 Services          0      8 KB
System                          4 Services          0    5.244 KB
Registry                       200 Services         0   36.012 KB
smss.exe                       604 Services         0     660 KB
csrss.exe                      760 Services         0    2.804 KB
wininit.exe                   848 Services         0    3.916 KB
services.exe                  920 Services         0   12.756 KB
lsass.exe                     940 Services         0   22.104 KB
svchost.exe                   700 Services         0   30.980 KB
fontdrvhost.exe              740 Services         0    1.560 KB
svchost.exe                   1040 Services        0   19.848 KB
svchost.exe                   1088 Services        0    6.364 KB
svchost.exe                   1388 Services        0    6.164 KB
svchost.exe                   1396 Services        0    5.552 KB
svchost.exe                   1504 Services        0   10.504 KB
svchost.exe                   1532 Services        0    2.272 KB
svchost.exe                   1564 Services        0    6.448 KB
svchost.exe                   1656 Services        0   13.920 KB
svchost.exe                   1704 Services        0   11.060 KB
svchost.exe                   1756 Services        0    3.904 KB
svchost.exe                   1864 Services        0    8.684 KB
NVIDIA.Container.exe          1916 Services        0   13.828 KB
svchost.exe                   1732 Services        0   15.948 KB
```

2. Lista sólo los que están en ejecución

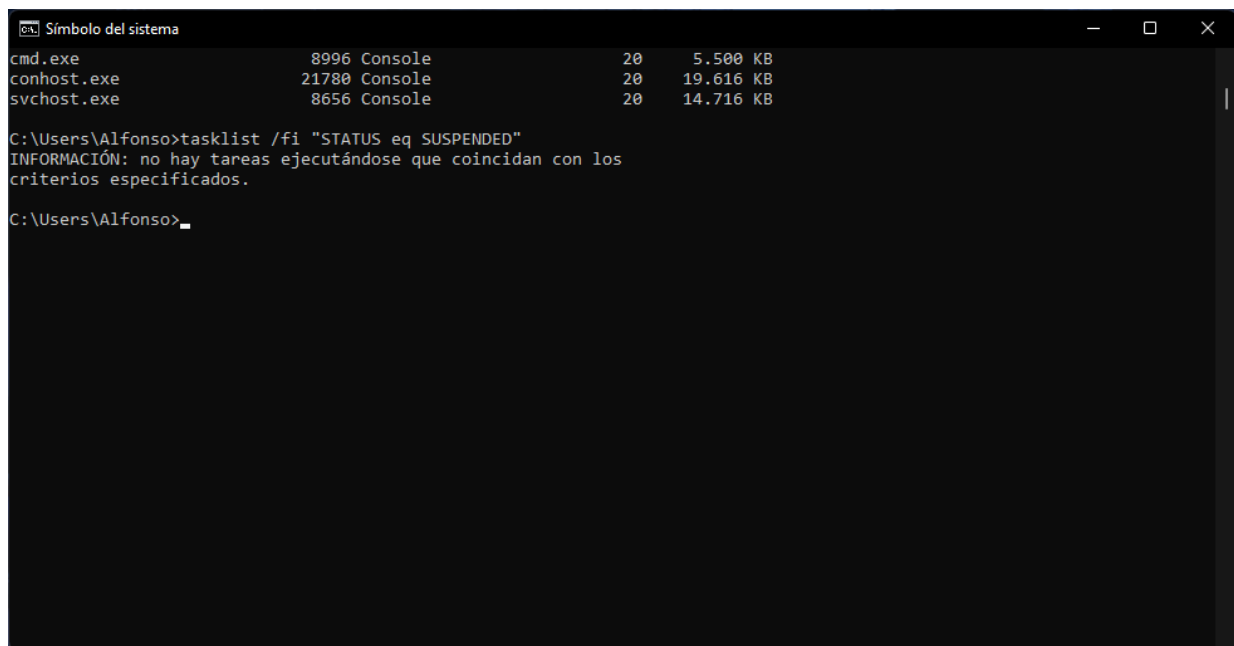
```
Símbolo del sistema

tasklist.exe                   25832 Console        20    9.480 KB
WmiPrvSE.exe                  6768 Services         0   10.192 KB

C:\Users\Alfonso>tasklist /fi "STATUS eq running"

Nombre de imagen                PID Nombre de sesión Núm. de ses Uso de memor
=====
csrss.exe                      23520 Console        20    7.196 KB
dwm.exe                       15476 Console        20   95.112 KB
NVIDIA.Container.exe          1452 Console        20   71.924 KB
nvcontainer.exe              3452 Console        20   33.128 KB
sihost.exe                   6080 Console        20   32.604 KB
svchost.exe                   8732 Console        20   36.996 KB
nvcontainer.exe             10324 Console        20   59.980 KB
explorer.exe                 26224 Console        20  200.000 KB
taskhostw.exe                19660 Console        20   17.280 KB
svchost.exe                  25608 Console        20   23.776 KB
StartMenuExperienceHost.exe  22432 Console        20   92.424 KB
SearchHost.exe               9500 Console        20  211.108 KB
RuntimeBroker.exe            17336 Console        20   29.012 KB
RuntimeBroker.exe            18672 Console        20   52.736 KB
dllhost.exe                  7960 Console        20   17.520 KB
YourPhone.exe                24820 Console        20   17.172 KB
ctfmon.exe                   20520 Console        20   21.908 KB
LockApp.exe                  27604 Console        20   60.320 KB
RuntimeBroker.exe            26980 Console        20   34.868 KB
NVIDIA Web Helper.exe         2660 Console        20   16.792 KB
nvshelper64.exe              18020 Console        20   14.608 KB
NVIDIA.Share.exe              9936 Console        20   75.084 KB
SecurityHealthSystray.exe     17556 Console        20   10.580 KB
```

3. Lista sólo los que están suspendidos

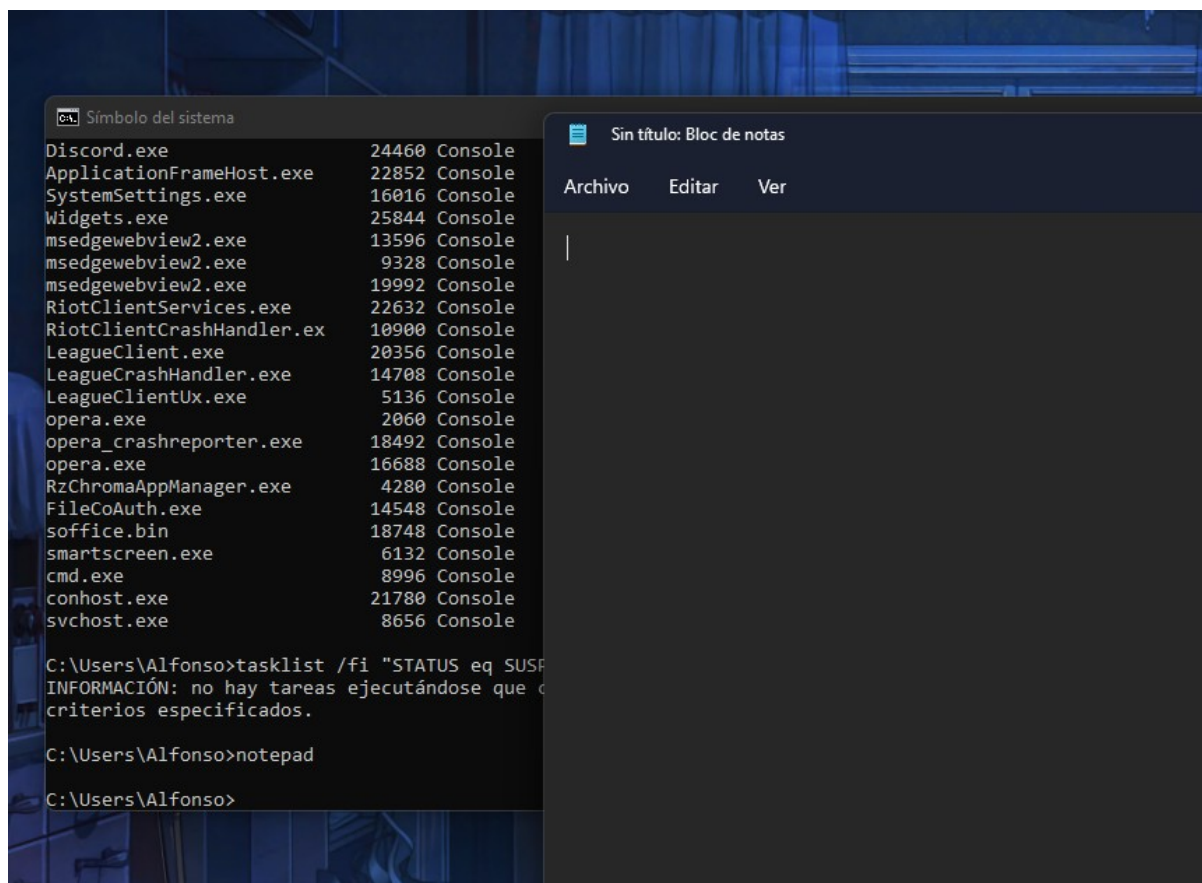


```
Símbolo del sistema
cmd.exe                8996 Console                20      5.500 KB
conhost.exe            21780 Console               20     19.616 KB
svchost.exe             8656 Console                20     14.716 KB

C:\Users\Alfonso>tasklist /fi "STATUS eq SUSPENDED"
INFORMACIÓN: no hay tareas ejecutándose que coincidan con los
criterios especificados.

C:\Users\Alfonso>
```

4. Abre un notepad en Windows desde línea de comandos



```
Símbolo del sistema
Discord.exe            24460 Console
ApplicationFrameHost.exe 22852 Console
SystemSettings.exe    16016 Console
Widgets.exe           25844 Console
msedgewebview2.exe    13596 Console
msedgewebview2.exe    9328 Console
msedgewebview2.exe    19992 Console
RiotClientServices.exe 22632 Console
RiotClientCrashHandler.exe 10900 Console
LeagueClient.exe      20356 Console
LeagueCrashHandler.exe 14708 Console
LeagueClientUx.exe    5136 Console
opera.exe              2060 Console
opera_crashreporter.exe 18492 Console
opera.exe              16688 Console
RzChromaAppManager.exe 4280 Console
FileCoAuth.exe        14548 Console
soffice.bin            18748 Console
smartscreen.exe        6132 Console
cmd.exe                8996 Console
conhost.exe            21780 Console
svchost.exe            8656 Console

C:\Users\Alfonso>tasklist /fi "STATUS eq SUSPENDED"
INFORMACIÓN: no hay tareas ejecutándose que coincidan con los
criterios especificados.

C:\Users\Alfonso>notepad

C:\Users\Alfonso>
```


5. Filtra el listado de procesos para obtener sólo información del notepad

```
Seleccionar Símbolo del sistema
opera_crashreporter.exe 18492 Console 20 9.880 KB
opera.exe 16688 Console 20 161.872 KB
RzChromaAppManager.exe 4280 Console 20 9.512 KB
FileCoAuth.exe 14548 Console 20 26.176 KB
soffice.bin 18748 Console 20 386.360 KB
smartscreen.exe 6132 Console 20 24.020 KB
cmd.exe 8996 Console 20 5.500 KB
conhost.exe 21780 Console 20 19.616 KB
svchost.exe 8656 Console 20 14.716 KB

C:\Users\Alfonso>tasklist /fi "STATUS eq SUSPENDED"
INFORMACIÓN: no hay tareas ejecutándose que coincidan con los
criterios especificados.

C:\Users\Alfonso>notepad

C:\Users\Alfonso>tasklist /fi "STATUS eq notepad.exe"
Error: No se reconoce el filtro de búsqueda.

C:\Users\Alfonso>tasklist /fi "IMAGENAME eq notepad.exe"
INFORMACIÓN: no hay tareas ejecutándose que coincidan con los
criterios especificados.

C:\Users\Alfonso>tasklist /fi "IMAGENAME eq notepad.exe"

Nombre de imagen PID Nombre de sesión Núm. de ses Uso de memor
=====
Notepad.exe 25508 Console 20 68.448 KB

C:\Users\Alfonso>
```

6. Finaliza el notepad filtrando por nombre

```
C:\Users\Alfonso>taskkill /fi "IMAGENAME eq notepad.exe"
Correcto: se envió la señal de término al proceso con PID 25508.

C:\Users\Alfonso>
```

7. Lanza el notepad con prioridad "por debajo de normal"

```
C:\Users\Alfonso>wmic process where name="notepad.exe" CALL setpriority normal
Ejecutando (\\DESKTOP-MA61SC7\ROOT\CIMV2:Win32_Process.Handle="22988")->setpriority()
Ejecución correcta del método.
Parámetros de salida:
instance of __PARAMETERS
{
    ReturnValue = 0;
};

C:\Users\Alfonso>
```

8. Cambia la prioridad del notepad anterior a Normal

```
C:\Users\Alfonso>wmic process where name="notepad.exe" CALL setpriority "Above normal"
Ejecutando (\\DESKTOP-MA61SC7\ROOT\CIMV2:Win32_Process.Handle="22988")->setpriority()
Ejecución correcta del método.
Parámetros de salida:
instance of __PARAMETERS
{
    ReturnValue = 0;
};

C:\Users\Alfonso>
```

9. Finaliza el notepad filtrando por PID

```
C:\Users\Alfonso>taskkill /f /PID 22988
Correcto: se terminó el proceso con PID 22988.

C:\Users\Alfonso>
```

Ejercicio B.3

Crea un script de bash con las siguientes características:

1. Recibe un único argumento, que puede ser: CPU, MEM o TIME (si el argumento falta, o se le pasan varios argumentos o no son válidos, entonces por defecto usará la opción CPU).
2. Si el usuario intenta interrumpir o matar (terminar) el script cuando esté en ejecución, se mostrará un mensaje diciendo que no es posible, que elija la opción Salir del menú.
3. Obtiene el proceso que MÁS CONSUMO tenga de %CPU, %MEM o TIME, según se haya indicado en el argumento. PISTA: para obtener esta información basta con usar un comando similar al siguiente, donde la variable `c` guarda la columna que se va a coger como referencia para ordenar (según el formato de `ps aux`, así la %CPU es la columna 3, %MEM es la columna 4 y TIME es la 10):

```
info=(`ps aux | tr -s " " | tail -n +2 | sort -hrk $c | head -n 1`)
```


De esta forma, `info` será un array con todos los datos que ofrece `ps aux`. Por ejemplo, el PID está en `${info[1]}`, el usuario en `${info[0]}`, el estado en `${info[7]}`, etc.
4. Luego muestra un menú que se repite hasta que el usuario indica que quiere salir. Antes de las opciones se mostrará la información sobre ese proceso, como usuario, PID, %CPU, %MEM, estado, tiempo, etc. (si usaste el comando anterior, ya tienes esta información en el array "info"). Las opciones del menú (se deben implementar las operaciones) se mostrarán a continuación, y son:
 1. Interrumpir el proceso
 2. Terminar el proceso
 3. Matar el proceso
 4. Detener el proceso y pasarlo a segundo plano
 5. Reanudar el proceso en segundo plano
 6. Pasar el proceso a segundo plano
 7. Asignarle la peor prioridad posible
 8. Evitar que el proceso muera si se cierra la shell donde se ejecuta.
 9. Salir

PISTA: Las señales que se pueden usar son: SIGCONT, SIGINT, SIGLKILL, SIGSTOP (o SIGTSTP), SIGTERM

Servicios

Ejercicio C.1

Resuelve los siguientes apartados en GNU/Linux. Cuando se pregunte por qué comando se utilizaría, debes indicar el comando completo, con las opciones y parámetros necesarios para que funcione:

1. Muestra todos los servicios de tu sistema

`service --status-all`

```
alumno@alfonsogj:~$ service --status-all
[ + ] acpid
[ - ] alsa-utils
[ - ] anacron
[ - ] apache-htcacheclean
[ + ] apache2
[ + ] apparmor
[ + ] apport
[ + ] avahi-daemon
[ - ] bluetooth
[ - ] console-setup.sh
[ + ] cron
[ + ] cups
[ + ] cups-browsed
[ + ] dbus
[ - ] docker
[ + ] gdm3
[ - ] grub-common
[ + ] hddtemp
[ - ] hwclock.sh
[ - ] irqbalance
[ + ] kerneloops
[ - ] keyboard-setup.sh
[ + ] kmod
[ - ] lightdm
[ + ] lm-sensors
[ + ] mysql
[ + ] network-manager
[ - ] open-vm-tools
[ - ] plymouth
[ - ] plymouth-log
[ - ] pppd-dns
[ + ] procps
[ - ] pulseaudio-enable-autospawn
[ - ] rsync
[ + ] rsyslog
[ - ] saned
[ - ] screen-cleanup
[ - ] speech-dispatcher
[ - ] spice-vdagent
```

2. ¿Cómo sabes si el servicio está activo o no?

Con el + estan activos y el – no lo estan. Debemos hacer un `service --status-all | grep +`

3. Elige 3 servicios de la lista y, para cada uno de ellos, indica:

1. El nombre
2. Investiga un poco para qué se usa
3. Si está activo o no (¿qué significa esto?)
4. Si está habilitado o no (¿qué significa esto?)
5. Si está enmascarado o no (¿qué significa esto?)

Los servicios que elijo que están activados son: cron, cups y dbus.

El cron es el administrador de tareas de Linux que permite ejecutar comandos en un momento determinado.

CUPS significa “Common UNIX Printing System” : Sistema común de impresión UNIX, permite que un computador actúe como servidor de impresión.

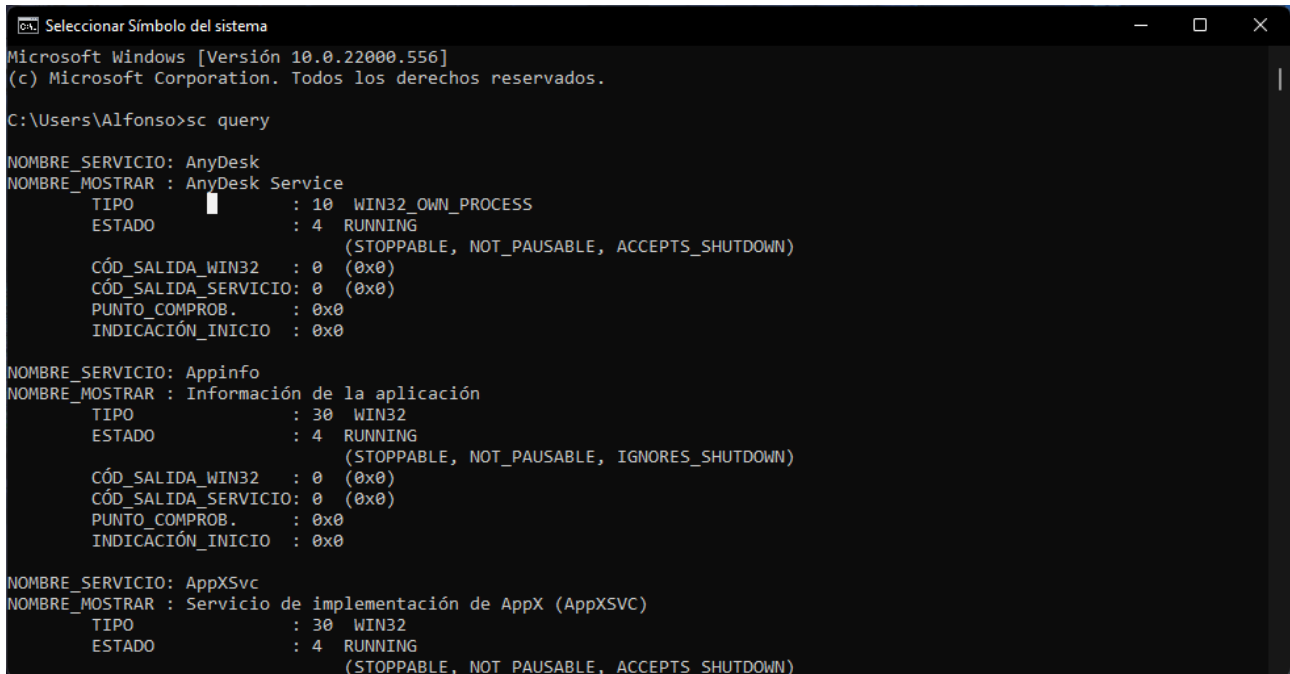
Es un sistema de comunicación entre procesos (IPC) y una llamada a procedimiento remoto (RPC) , para aplicaciones de software con el fin de comunicarse entre sí.

Enmascarar un servicio evita que el servicio se inicie manual o automáticamente.

Ejercicio C.2

a) Elige un servicio no primordial de Windows y usa comandos `sc` para obtener datos de este servicio y de su estado actual. Prueba a detenerlo, iniciarlo, pausarlo, reanudarlo, etc. Déjalo en su estado inicial.

Se realiza un `sc query` para elegir el servicio.



```
Microsoft Windows [Versión 10.0.22000.556]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\Alfonso>sc query

NOMBRE_SERVICIO: AnyDesk
NOMBRE_MOSTRAR : AnyDesk Service
        TIPO           : 10  WIN32_OWN_PROCESS
        ESTADO          : 4   RUNNING
                        (STOPPABLE, NOT_PAUSABLE, ACCEPTS_SHUTDOWN)
        CÓD_SALIDA_WIN32 : 0   (0x0)
        CÓD_SALIDA_SERVICIO: 0   (0x0)
        PUNTO_COMPROB.   : 0x0
        INDICACIÓN_INICIO : 0x0

NOMBRE_SERVICIO: Appinfo
NOMBRE_MOSTRAR : Información de la aplicación
        TIPO           : 30  WIN32
        ESTADO          : 4   RUNNING
                        (STOPPABLE, NOT_PAUSABLE, IGNORES_SHUTDOWN)
        CÓD_SALIDA_WIN32 : 0   (0x0)
        CÓD_SALIDA_SERVICIO: 0   (0x0)
        PUNTO_COMPROB.   : 0x0
        INDICACIÓN_INICIO : 0x0

NOMBRE_SERVICIO: AppXSvc
NOMBRE_MOSTRAR : Servicio de implementación de AppX (AppXSVC)
        TIPO           : 30  WIN32
        ESTADO          : 4   RUNNING
                        (STOPPABLE, NOT_PAUSABLE, ACCEPTS_SHUTDOWN)
```

Y usamos los siguientes comandos:

Iniciarlo: `sc start servicio`

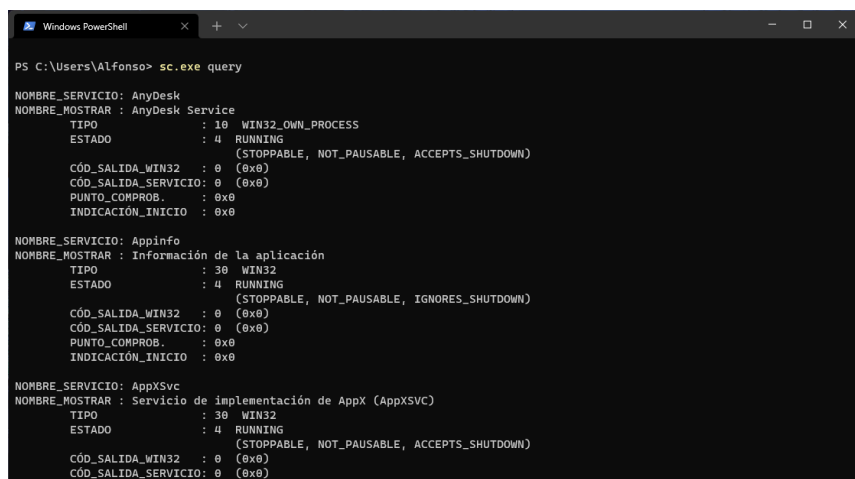
Detenerlo: `sc stop servicio`

Pausar: `sc pause servicio`

Reiniciar: `sc reload servicio`

b) Repite el apartado anterior usando PowerShell

El comando sería `sc.exe query`



```
PS C:\Users\Alfonso> sc.exe query

NOMBRE_SERVICIO: AnyDesk
NOMBRE_MOSTRAR : AnyDesk Service
        TIPO           : 10  WIN32_OWN_PROCESS
        ESTADO          : 4   RUNNING
                        (STOPPABLE, NOT_PAUSABLE, ACCEPTS_SHUTDOWN)
        CÓD_SALIDA_WIN32 : 0   (0x0)
        CÓD_SALIDA_SERVICIO: 0   (0x0)
        PUNTO_COMPROB.   : 0x0
        INDICACIÓN_INICIO : 0x0

NOMBRE_SERVICIO: Appinfo
NOMBRE_MOSTRAR : Información de la aplicación
        TIPO           : 30  WIN32
        ESTADO          : 4   RUNNING
                        (STOPPABLE, NOT_PAUSABLE, IGNORES_SHUTDOWN)
        CÓD_SALIDA_WIN32 : 0   (0x0)
        CÓD_SALIDA_SERVICIO: 0   (0x0)
        PUNTO_COMPROB.   : 0x0
        INDICACIÓN_INICIO : 0x0

NOMBRE_SERVICIO: AppXSvc
NOMBRE_MOSTRAR : Servicio de implementación de AppX (AppXSVC)
        TIPO           : 30  WIN32
        ESTADO          : 4   RUNNING
                        (STOPPABLE, NOT_PAUSABLE, ACCEPTS_SHUTDOWN)
        CÓD_SALIDA_WIN32 : 0   (0x0)
        CÓD_SALIDA_SERVICIO: 0   (0x0)
```

Ejercicio C.3

Crear un script en bash que reciba un único argumento (obligatorio) que debe ser el nombre de un servicio. A partir de este nombre:

- 1. Si el servicio no está en el sistema, da un error y termina la ejecución*
- 2. Si el servicio está en el sistema, se mostrará su estado (si está activo/inactivo, habilitado/deshabilitado o enmascarado/no enmascarado)*
- 3. Luego se mostrarán tres preguntas para cambiar el estado. Por ejemplo, el servicio está activo, ¿desea usted pasarlo a inactivo? (o viceversa, según el estado). Preguntar lo mismo para cambiar el estado de habilitado y enmascarado*