



Instituto Tecnológico de Costa Rica

Curso: Microcontroladores y microprocesadores

Profesor: Rodolfo Piedra Camacho

Tarea 2

Estudiantes:

Xochilth Víctor Angulo (2019190056)

Marco Rodríguez Morales (2019163031)

Carlos Cortés Chacón (2018118462)

Álvaro Quirós Hernández (2018093990)

Grupo: 01

I Semestre

2022

Fecha de Entrega: 11/03/2022

## Preguntas Teóricas

### 1. ¿Qué es el Shell de Linux?

El shell de Linux es un intérprete de comandos basados en Linux y que de cierta manera se puede decir que es su interfaz de usuario original. Con esto es posible realizar tareas directamente desde el núcleo y de esta forma controlar el funcionamiento de la computadora. [1] [2]

### 2. ¿Qué es gcc?

GCC es un compilador que se ejecuta en la mayoría de las plataformas y además, tiene la capacidad de producir salidas para la mayoría de los procesadores disponibles tales como los procesadores para computadoras, microcontroladores y CPUs de 64 bits. [3]

### 3. ¿Cuál es la diferencia entre gcc y g++?

GCC	G++
Se usa para compilar C	Se usa para compilar C++
Puede correr programas de tipo .c o .cpp pero los va a tratar respectivamente si son C o C++	Puede correr programas de tipo .c o .cpp pero los va a tratar siempre como si fuera C ++
No puede realizar esta acción	realiza de manera automática el link de archivos a las librerías std de C++

### 4. ¿Cómo se compila un programa en C utilizando el compilador gcc desde la terminal de Linux? Brinde un ejemplo.

Para compilar un programa C en GCC se utiliza el siguiente comando: gcc fileName.c -o binary. [4]

#### Ejemplo:

Se debe de hacer en varios pasos. Se debe de tener un código:  
#include<stdio.h>

```
int main(void)
{
    printf("hello\n");
    return 0;
}
```

Luego, se debe de hacer un proceso de precompilación:  
gcc -E -I . main.c -o main.i

# -E es hacer que el compilador salga después del preprocesamiento  
# -I Especifico el directorio del archivo de encabezado  
# -o Especificque el nombre del archivo de salida

Luego, se expande el código con información:  
# 1 "main.c"  
# 1 "<built-in>"

```
# 1 "<línea de comando>"
# 1 "/usr/include/stdc-predef.h" 1 3 4
# 1 "<Línea de comando>" 2
# 1 "main.c"
# 1 "/usr/include/stdio.h" 1 3 4
...

# 28 "/usr/include/bits/types.h" 2 3 4
...

typedef unsigned char __u_char;
typedef unsigned short int __u_short;
typedef unsigned int __u_int;
typedef unsigned long int __u_long;
...
```

Luego, simplemente se utiliza el código para compilar y queda de la siguiente manera:

```
gcc -S -I . main.i -o main.s
# -S Deje que el compilador se detenga después de la compilación.
```

## 5. ¿Cómo se ejecuta un programa en C ya compilado desde la terminal de Linux?

Luego de haber compilado el programa, simplemente basta con volver a cmd y escribir el nombre del programa para que luego este se ejecute.

## 6. ¿Qué es el comando ifconfig de Ubuntu? ¿Qué información provee?

Este comando se usa para configurar los interfaces de red residentes en el núcleo. Se utiliza durante el arranque para configurar aquellos interfaces que se necesiten. Tras esto, normalmente sólo se necesita para la depuración de errores o para la puesta a punto del sistema [5]. A través de este comando se obtiene información sobre las interfaces de un sistema específico. Por ejemplo, se puede obtener información sobre [6]:

- Nombres de dispositivo de todas las interfaces de un sistema
- Todas las direcciones IPv4 y, si las hay, IPv6 asignadas a las interfaces
- Si estas interfaces están configuradas

## 7. ¿Para qué sirve el protocolo SSH?

El Secure Shell (SSH) es un protocolo que tiene como función ofrecer acceso remoto a un servidor. La principal peculiaridad es que este acceso es seguro, ya que toda la información va cifrada. Esto evita que pueda filtrarse y que un tercero pueda ver esos datos. [7]

## 8. ¿Qué es la herramienta openssh-server de Linux?

OpenSSH es la versión de código abierto de las herramientas de Secure Shell (SSH) que usan los administradores de Linux y otras que no son de Windows para la administración de distintas plataformas de sistemas remotos. [8]

## 9. Describa las diferencias entre las tarjetas basadas en microprocesador Raspberry Pi 2 y Raspberry Pi 3.

Algunas diferencias son [9]:

- En Raspberry Pi 3, los tiempos de arranque son altos con diferentes dispositivos, mientras que los tiempos de arranque son menores en Raspberry Pi 2.
- En Raspberry Pi 3, el tiempo de instalación del software es menor, mientras que el tiempo de instalación del software es mayor en la versión Raspberry Pi 2.
- En Raspberry Pi 3, el SoC a menudo se calienta fácilmente y requiere algunos agentes de enfriamiento para controlar los niveles de temperatura, mientras que en Raspberry Pi 2 no se requiere un radiador ya que las temperaturas en SoC no alcanzan altos niveles fácilmente.
- En Raspberry Pi 3, los dispositivos periféricos a menudo se pueden conectar mediante conectividad inalámbrica, mientras que en Raspberry Pi 2, los dispositivos periféricos se pueden conectar a través de concentradores USB con alimentación.
- En Raspberry Pi 3, la capacidad de representación de cuadros por segundo es de 37.9 FPS, mientras que la velocidad de cuadros por segundo en Raspberry Pi 2 es de 25.4 FPS.
- En Raspberry Pi 3, la prueba de velocidad de red con LAN ofrece la mejor velocidad en términos de Ethernet inactivo y baja en términos de Ethernet Up, mientras que Raspberry Pi 2 tiene la misma velocidad en términos de USB WLAN Up en comparación con Raspberry Pi 3. En términos de Internet WLAN también, Raspberry Pi 3 tiene el mejor rendimiento.
- En Raspberry Pi 3, el aumento de rendimiento es mayor en comparación con Raspberry Pi 2. Raspberry Pi 3 es más rápido y tiene varios beneficios de aumento de rendimiento en comparación con Raspberry Pi 2.
- En Raspberry Pi 3, la tarjeta Micro SD es un mecanismo basado en fricción en lugar de un tipo push-push con resorte que se ha utilizado hasta la versión Raspberry Pi 2.
- En Raspberry Pi 3, la prueba de rendimiento con la herramienta de evaluación comparativa Whetstone Pi A7 reveló el puntaje de 711, mientras que el puntaje de Raspberry Pi 2 es 432, lo que indica claramente que Raspberry Pi 3 tiene más beneficios.
- En Raspberry Pi 3, el consumo de energía inactiva es menor, es decir, 2.5 W, mientras que el consumo de energía inactiva en Raspberry Pi 2 es 3.2 W, que es más.
- En Raspberry Pi 3, la potencia bajo carga es de 3.8 W, que es la misma en el caso de las versiones Raspberry Pi 3 y Raspberry Pi 2.
- En Raspberry Pi 3, el consumo de energía es menor, la velocidad de la CPU es más rápida, el procesador central con gráficos más rápidos y con Wi-Fi y Bluetooth incorporados que carecen de todas estas características en la versión Raspberry Pi 2, que es un impulso de alto rendimiento para la última versión y sus usuarios. En Raspberry Pi 3, la prueba de memoria para la RAM para el límite establecido de 256 MB del rendimiento de la velocidad, el tiempo tomó 8 minutos, mientras que en el modelo Raspberry Pi 2 el tiempo necesario fue de 22 minutos, que es más.
- En Raspberry Pi 3, el procesador ARM Cortex de 64 bits admite el sistema operativo de diferentes maneras en comparación con las versiones anteriores, como Raspberry Pi 2.

**10. Defina los siguientes conceptos de programación por hilos: Thread, lock, join y deadlock**

- **Thread:** Es la menor de las estructuras lógicas de programación que se ejecuta de forma secuencial por parte del planificador del sistema operativo. En otras palabras, el thread o hilo es simplemente una tarea que puede ser ejecutada al mismo tiempo que otra tarea. Los hilos de ejecución que comparten los mismos recursos, sumados a estos recursos, son en conjunto conocidos como un proceso. [10]
- **Lock:** Los objetos Lock permiten gestionar los bloqueos que evitan que los hilos modifiquen variables compartidas al mismo tiempo. El método `acquire()` permite que un hilo bloquee a otros hilos en un punto del programa, donde se leen y actualizan datos, hasta que dicho hilo libere el bloqueo con el método `release()`. En el momento que se produzca el desbloqueo otro hilo (o el mismo) podrá bloquear de nuevo. [11]
- **Join:** El método `join()` se utiliza para mantener la ejecución del hilo que se está ejecutando actualmente hasta que el hilo especificado esté muerto (ejecución terminada). [12]
- **Deadlock:** también llamado interbloqueo punto muerto. Este describe varios subprocesos están bloqueados al mismo tiempo, uno o más de ellos están esperando que se libere un determinado recurso, pero este recurso está ocupado por el otro, lo que hace que los subprocesos se bloqueen indefinidamente, el programa no se puede ejecutar normalmente y el programa no se puede terminar normalmente. [13]

## Referencias bibliográficas

- [1] J. Manjarrez, “¿Que es un Shell?” [Online]. Available: <https://blog.desdelinux.net/que-es-un-shell/>. [Accessed: 2022]
- [2] C. Lopez, “¿Qué es la ‘shell’ de Linux?” [Online]. Available: <https://es.ccm.net/contents/316-linux-shell>. [Accessed: 2022]
- [3] “Difference between GCC and G++,” 2022. [Online]. Available: <https://www.geeksforgeeks.org/difference-between-gcc-and-g/>. [Accessed: 2022]
- [4] “Proceso de compilación del programa y compilador GCC.” [Online]. Available: [https://programmerclick.com/article/42311144850/#GCC\\_\\_64](https://programmerclick.com/article/42311144850/#GCC__64). [Accessed: 2022]
- [5] *Ubuntu Manpage: ifconfig - configura un interfaz de red.* (2019). Ubuntu Manpage:Welcome. <http://manpages.ubuntu.com/manpages/bionic/es/man8/ifconfig.8.html#:~:text=Ifconfig%20se%20usa%20para%20configurar,puesta%20a%20punto%20del%20sistema>
- [6] Oracle Corporation and/or its affiliates. (2010). *Supervisión de la configuración de interfaz con el comando ifconfig (Guía de administración del sistema: servicios IP)*. Oracle. <https://docs.oracle.com/cd/E19957-01/820-2981/ipconfig-141/index.html>
- [7] Jiménez, J. (2021, 13 de agosto). *Qué es y para qué sirve el SSH*. RedesZone. <https://www.redeszone.net/tutoriales/internet/protocolo-ssh-usos/>
- [8] Maertens, D. y Olprod. (2021, 12 de agosto). *Información general sobre OpenSSH para Windows*. Developer tools, technical documentation and coding examples | Microsoft Docs. <https://docs.microsoft.com/es-es/windows->

[server/administration/openssh/openssh\\_overview#:~:text=OpenSSH%20es%20la%20versi%20de,distintas%20plataformas%20de%20sistemas%20remotos](#)

[9] *Raspberry Pi 3 vs Raspberry Pi 2 - Conozca los 8 Comparación útil.* (2022). Gráficos, Diseño, Cálculo, <https://es.education-wiki.com/2008926-raspberry-pi-3-vs-raspberry-pi-2>

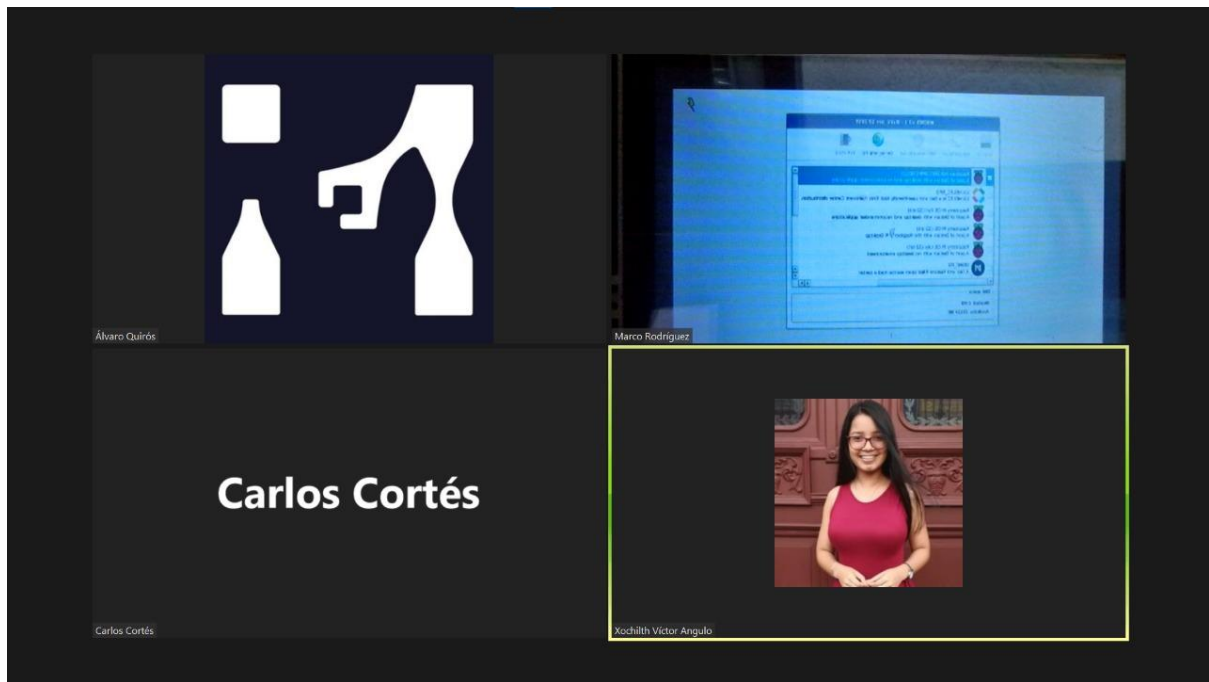
[10] *¿qué son los hilos en la programación?* (2021). ALEP. <https://aleph.org.mx/que-son-los-hilos-en-la-programacion>

[11] Suárez Lamadrid, A. y Suárez Jiménez, A. (s. f.). *Threading: programación con hilos (y II).* Python 3 para impacientes. [https://python-para-impacientes.blogspot.com/2016/12/threading-programacion-con-hilos-y-ii.html#:~:text=Los%20objetos%20Lock%20permiten%20gestionar,con%20el%20método%20release\(\)](https://python-para-impacientes.blogspot.com/2016/12/threading-programacion-con-hilos-y-ii.html#:~:text=Los%20objetos%20Lock%20permiten%20gestionar,con%20el%20método%20release())

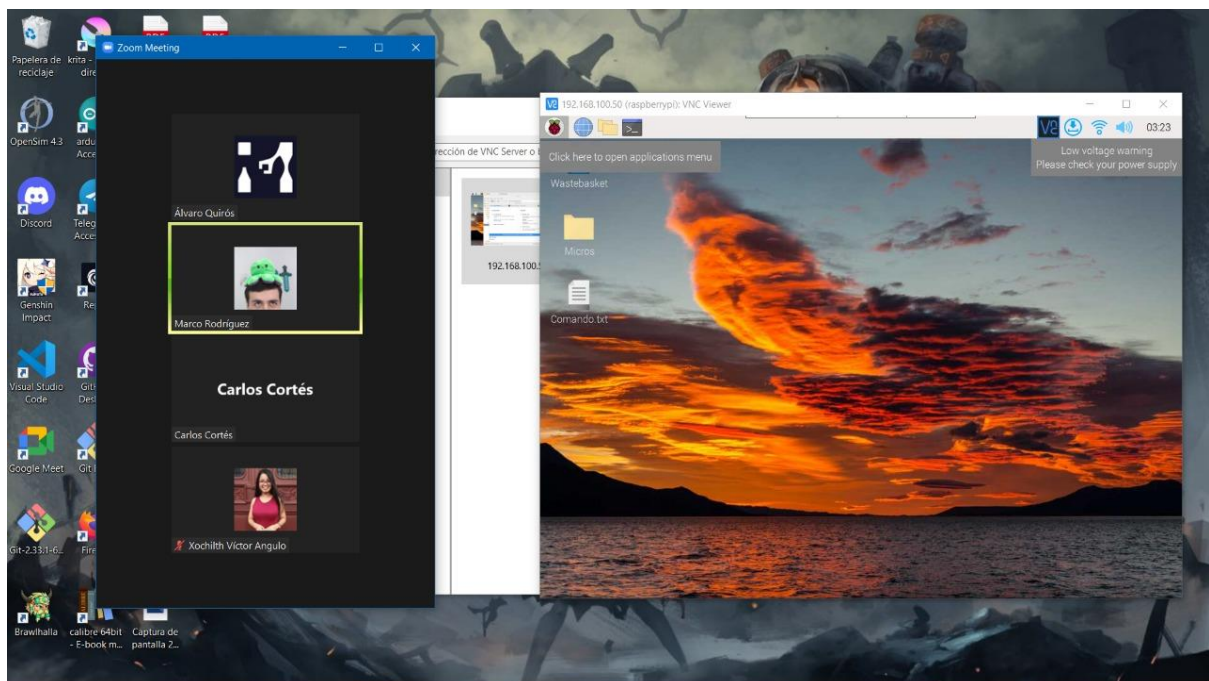
[12] *El método join() en Java con el ejemplo - Tecnologías Móviles.* (2022). Tecnologías Móviles. [https://tecnologiasmoviles.com/el-metodo-join-en-java-con-el-ejemplo/#:~:text=El%20método%20join\(\)%20se,esté%20muerto\(ejecución%20terminada\)](https://tecnologiasmoviles.com/el-metodo-join-en-java-con-el-ejemplo/#:~:text=El%20método%20join()%20se,esté%20muerto(ejecución%20terminada))

[13] *¿Qué es el interbloqueo de hilo? - programador clic.* (s. f.). programador clic. <https://programmerclick.com/article/42411673896/>

## Capturas de pantalla



**Figura 1.** Construcción de la imagen Raspbian Noobs



**Figura 2.** Construcción de la imagen Debian Bullseye