

Examen Parcial
Sistemas operativos
Primavera 2026

Sección Teórica

- 1.- ¿El modo usuario tiene acceso sin restricciones al hardware?
 - a) Verdadero
 - b) Falso**
- 2.- ¿Qué virtualiza el sistema operativo?
 - a) CPU
 - b) MEMORIA**
 - c) DISCO DURO
- 3.- ¿Los procesos usan _____ para comunicar se con el kernel?
 - a) System Calls**
 - b) Librerías
 - c) Hilos
- 4.- ¿Tres componentes del sistema operativo para administrar la memoria, dispositivos, etc.?
 - a) Virtualización, Persistencia, Concurrencia
 - b) Procesos, Address Space, Stack**
 - c) Sistema de archivos, Hardware, Heap
- 5.- ¿Cuáles son los tres estados de un proceso?
 - a) Running, Blocked, Ready**
 - b) Fork, Wait, Exe
 - c) Joined, Detach, Yielded
- 6.- ¿Algoritmo que usa time-slices para simular paralelismo en procesos?
 - a) Round-Robin**
 - b) FIFO
 - c) Shortest Job First
- 7.- ¿El Address space de un proceso es compartido entre los hilos del mismo proceso?
 - a) Verdadero**
 - b) Falso

8.- ¿Los procesos terminan voluntariamente cuando ____?

- a) Tiene salida por error
- b) Tiene un error fatal**
- c) Por instrucción de un proceso padre

9.- ¿Sólo los programas en C tienen Stack y Heap?

- a) Verdadero
- b) Falso**

10.- ¿Los primitivos son sólo parte del estándar de C y C++?

- a) Verdadero
- b) Falso**

11.- ¿Los schedulers no tienen en cuenta la duración de los procesos por ____?

- a) Las interrupciones mismas del sistema no dejan determinar un tiempo definido
- b) Los procesos no saben su propia duración en el sistema**
- c) El scheduler al hacer context switching no determina el tiempo de vida de un proceso

12.- ¿Los mutex sólo sirven para regular las race conditions?

- a) Verdadero
- b) Falso**

13.- ¿Qué formas hay de comunicación entre hilos?

- a) Sockets, Condition variables, Memoria compartida**
- b) Sockets, Condition Variables, Paso por mensaje
- c) Sockets, Atomic Variables, Promises

14.- ¿La función join() se puede llamar más de dos veces en un mismo hilo?

- a) Verdadero
- b) Falso**

15.- ¿Qué es el user space?

Es la memoria que el kernel otorga para los procesos del usuario, un proceso es un estado en la memoria, el user space es la memoria para el conjunto de procesos que se están ejecutando.

16.- ¿Cuáles son las partes del prompt de la terminal?

- name@host:path\$: el primero es el nombre de usuario seguido del arroba el nombre de la máquina y la ruta en la que nos encontramos parados.

Sección Práctica

1.-¿Qué realizaría este comando?

```
cd /etc; ls | grep cron
```

- Primer paso cambia el directorio a /etc, como segundo paso se lista todo lo que este en /etc y lo filtra con el comando grep para encontrar todo aquel archivo o carpeta que tenga las letras cron en su nombre.

2.-¿Qué realizaría este comando?

```
echo "alias code=codium" >> .bashrc
```

- Lo que está creando aquí es un alias que va a afectar a la shell bash para que cuando se escriba la palabra code se abra el programa codium y lo está agregando usando redirección hacia .bashrc

3.-¿Qué realizaría este comando?

```
echo $PATH; export PATH=/usr/bin/vim:$PATH
```

Aquí el comando se utiliza para crear algo similar a un acceso directo temporal cambiando la ruta actual usando export para otorgarle otra. Lo que provoca que este path tenga prioridad.

4. Realizar un programa en C++ que escriba en un archivo de texto plano usando multithreading, cada thread tiene que escribir algo diferente y en el archivo donde se escribe se tiene que ver el texto de forma correcta, cada línea tiene que ser el texto de un thread diferente. Usen 5 threads diferentes para resolver el problema.

5. Ejecuten un one liner(una línea de un solo comando) que primero cree un archivo de texto plano llamado accept.txt, después de crearlo liste los contenidos del directorio y redirija su salida para escribir en este archivo de texto, después liste todos los comandos utilizados hasta ahora en su sesión de terminal, encuentre las veces que se repite en esta salida el comando ls y redirija la salida de ese comando al final del archivo accept.txt