

### 1. ¿Qué es una computadora? Indique sus componentes fundamentales.

Una computadora, sin software, es básicamente un conjunto de componentes electrónicos sin utilidad. Con el software, puede almacenar, procesar y recuperar información, mostrar documentos multimedia, realizar búsquedas en internet y llevar a cabo muchas otras tareas útiles.

El software de la computadora se divide en:

- **Programas de sistema:** controlan la operación del propio equipo.
- **Programas de aplicación:** realizan tareas específicas que el usuario desea ejecutar.

El programa más importante es el **sistema operativo**, que administra todos los recursos de la computadora y sirve como base para los programas de aplicación.

Los **componentes fundamentales** de una computadora incluyen:

- Uno o más procesadores.
- Memoria principal (RAM – Memoria de Acceso Aleatorio).
- Discos.
- Impresoras.
- Interfaces de red.
- Otros dispositivos de entrada y salida.

### 2. ¿Qué es el CPU?

El CPU (Central Processing Unit – Unidad Central de Procesamiento) es el componente que controla el funcionamiento de la computadora y realiza sus funciones de procesamiento de datos. Cuando existe un único procesador, se lo llama comúnmente CPU.

### 3. ¿Qué es la memoria principal? ¿Cuál es su función? Ejemplifique.

La memoria principal, también llamada **memoria real o primaria**, es donde se almacenan temporalmente los datos y programas que está utilizando el sistema.

- Es **volátil**, es decir, se borra cuando se apaga la computadora.

#### **Función:**

Proporcionar un espacio rápido de trabajo temporal para que la CPU acceda a los datos e instrucciones necesarias durante la ejecución de tareas.

#### **Ejemplos de uso:**

- Ejecución de programas.
- Multitarea.
- Manipulación de datos en tiempo real.

### 4. ¿Qué es la memoria secundaria? ¿Cuál es su función? Ejemplifique.

La **memoria secundaria** (o auxiliar) es una memoria no volátil externa, utilizada para almacenar grandes cantidades de datos de forma duradera.

- Suele presentarse en forma de **archivos y registros**.

**Función:**

Almacenar el sistema operativo, aplicaciones, archivos de usuario y otros datos que deben conservarse aún cuando el sistema se apague.

**Ejemplos:**

- Disco Duro (HDD).
- SSD.
- Unidades ópticas (DVD/Blu-ray).
- Dispositivos USB.

**5. ¿Qué es el almacenamiento fuera de línea?**

El **almacenamiento fuera de línea (offline storage)** se refiere a dispositivos que **no están conectados permanentemente** al sistema.

- Requieren intervención manual para su uso.
- Se utilizan comúnmente para copias de seguridad, archivado de información o transferencia de datos.

**Ejemplos:**

- Discos externos.
- Cintas magnéticas.
- DVDs.

**6. ¿Qué es la jerarquía de memorias?**

Es la estructura organizativa de los diferentes tipos de memoria en un sistema, ordenados según su **velocidad, capacidad y costo**.

- Los niveles más altos (como los registros y la memoria caché) son más rápidos y costosos.
- A medida que se baja en la jerarquía (RAM, disco duro, almacenamiento externo), la velocidad disminuye, pero la capacidad y la economía aumentan.

**Objetivo:**

Lograr un balance óptimo entre rendimiento y costo, aprovechando las ventajas de cada tipo de memoria.

**7. ¿Qué es la memoria caché?**

La **memoria caché** es un tipo de almacenamiento de alta velocidad que almacena datos e instrucciones **frecuentemente utilizados por la CPU**.

- Actúa como intermediario entre la CPU y la memoria principal.
- Mejora el rendimiento del sistema reduciendo los tiempos de acceso mediante los principios de **localidad temporal y espacial**.

**8. ¿Qué es un buffer?**

Un **buffer** es una zona de almacenamiento temporal utilizada para gestionar la

**transferencia de datos entre dispositivos que operan a diferentes velocidades.**

- Se usa comúnmente en operaciones de entrada/salida (E/S).
- Permite que los dispositivos funcionen sin requerir sincronización directa con la CPU.

**Ejemplo:**

Al imprimir un documento, los datos se almacenan temporalmente en un buffer antes de ser enviados a la impresora.

## **9. ¿Qué es la memoria virtual?**

La memoria virtual es una técnica que permite a los programas usar más memoria de la que hay físicamente disponible, mediante direcciones lógicas que se traducen dinámicamente en direcciones físicas, utilizando espacio en disco para cargar solo las partes necesarias de los procesos.

Nose cual:

La **memoria virtual** es una técnica que permite a los programas usar más memoria de la que hay físicamente disponible.

- Utiliza **espacio en disco** para simular memoria adicional.
- Permite a los procesos ejecutarse como si tuvieran acceso a una **gran memoria continua**, independientemente de la memoria física real.
- Las direcciones lógicas se traducen dinámicamente en direcciones físicas.

## **10. ¿Cuáles son los componentes fundamentales del CPU?**

Los componentes fundamentales de la Unidad Central de Procesamiento (CPU) incluyen:

- Unidad de control: controla el funcionamiento de la CPU y por tanto del computador.
- Unidad aritmético-lógica (ALU, Arithmetic Logic Unit): lleva a cabo las funciones de procesamiento de datos del computador.
- Registros: proporcionan almacenamiento interno a la CPU.
- Interconexiones CPU o buces internos: son mecanismos que proporcionan comunicación entre la unidad de control, la ALU y los registros.

Buces: (canales que transportan datos, direcciones y señales de control.)

## **11. ¿Cuál es la función de la UC (Unidad de Control)?**

La Unidad de Control (UC) es responsable de dirigir el funcionamiento del procesador. Controla la transferencia de datos e instrucciones dentro del procesador, así como entre el procesador y dispositivos externos. Además, interpreta las instrucciones y genera las señales de control necesarias para su ejecución.

## 12. ¿Cuál es la función de la ALU (Unidad Aritmético Lógica)?

La Unidad Aritmético-Lógica (ALU) realiza las operaciones aritméticas y lógicas del procesador. Opera sobre los datos proporcionados por los registros y produce resultados que se almacenan nuevamente en registros o en la memoria.

## 13. ¿Qué elemento de la UC indica la dirección de la siguiente instrucción a ejecutarse?

El **Contador de Programa (PC)** es el registro que contiene la dirección de la siguiente instrucción a ejecutar. Durante el ciclo de instrucción, el PC se actualiza para apuntar a la instrucción siguiente, a menos que una instrucción de control altere su contenido.

## 14. ¿Qué es el ciclo de vida de una instrucción y cuáles son sus fases?

El ciclo de vida de una instrucción, también conocido como ciclo de instrucción, comprende las siguientes fases:

1. **Captación (Fetch):** La CPU obtiene la instrucción desde la memoria, utilizando la dirección almacenada en el contador de programa.
2. **Decodificación (Decode):** La instrucción captada se interpreta para determinar qué acción se debe realizar.
3. **Ejecución (Execute):** La operación especificada por la instrucción se lleva a cabo.
4. **Acceso a memoria (Memory Access):** Si la instrucción requiere acceso a memoria, se realiza en esta fase.
5. **Escritura de resultado (Write Back):** El resultado de la ejecución se escribe en el destino correspondiente (registro o memoria).

Este ciclo se repite para cada instrucción del programa.

## 15. Explique qué son las técnicas de direccionamiento. Explique y distinga los modos directo, indirecto, inmediato y por base y desplazamiento.

Las técnicas de direccionamiento determinan cómo una instrucción especifica la ubicación de sus operandos. Los modos mencionados son:

- **Directo:** La dirección efectiva del operando está especificada directamente en la instrucción.

- **Indirecto:** La instrucción contiene una dirección que apunta a otra dirección donde se encuentra el operando.
- **Inmediato:** El operando es parte de la propia instrucción; no se requiere acceso adicional a memoria.
- **Por base y desplazamiento:** La dirección efectiva se calcula sumando un valor base (almacenado en un registro) y un desplazamiento especificado en la instrucción.

## 16. ¿Qué son los periféricos?

Los periféricos son dispositivos externos conectados al computador que permiten la entrada y salida de datos. Incluyen dispositivos como teclados, ratones, impresoras, monitores y unidades de almacenamiento externas.

## 17. Explique los mecanismos para tratamiento de E/S: Guiado por programa, por interrupciones y acceso directo a memoria.

Los mecanismos para el tratamiento de Entrada/Salida (E/S) son:

- **E/S guiada por programa:** El procesador controla directamente todas las operaciones de E/S, verificando constantemente el estado del dispositivo (polling).
- **E/S mediante interrupciones:** El dispositivo de E/S interrumpe al procesador cuando está listo para transferir datos, permitiendo que el procesador realice otras tareas mientras tanto.
- **Acceso Directo a Memoria (DMA):** Un controlador de DMA maneja la transferencia de datos entre la memoria y el dispositivo de E/S, liberando al procesador de esta tarea y mejorando la eficiencia del sistema.

TP2:

## 1. ¿Qué es un Sistema Operativo?

Un sistema operativo es un programa que actúa como intermediario entre el usuario y el hardware del computador. Su objetivo principal es proporcionar un entorno en el que el usuario pueda ejecutar programas de manera conveniente y eficiente. Además, gestiona los recursos del hardware y ofrece servicios a los programas de aplicación.

## 2. Describa los diferentes tipos de arquitecturas de sistemas operativos que conoce con sus ventajas y desventajas.

Stallings describe varias arquitecturas de sistemas operativos:

- **Monolítica:** Todos los componentes del sistema operativo están integrados en un solo programa que se ejecuta en modo kernel.
  - *Ventajas:* Alto rendimiento debido a la comunicación directa entre componentes.
  - *Desventajas:* Difícil de mantener y depurar; una falla en un componente puede afectar a todo el sistema.
- **Estructurada en capas:** El sistema operativo se divide en capas, cada una construida sobre la anterior.
  - *Ventajas:* Mejor organización y modularidad; facilita el mantenimiento.
  - *Desventajas:* Puede haber una sobrecarga en la comunicación entre capas.
- **Microkernel:** Solo las funciones esenciales residen en el kernel; otros servicios se ejecutan en espacio de usuario.
  - *Ventajas:* Mayor seguridad y estabilidad; los fallos en servicios no esenciales no afectan al núcleo.
  - *Desventajas:* Posible disminución del rendimiento debido a la comunicación entre procesos.
- **Cliente-servidor:** El sistema operativo se organiza como un conjunto de servicios que se comunican mediante mensajes.
  - *Ventajas:* Flexibilidad y escalabilidad; fácil distribución de servicios.
  - *Desventajas:* Complejidad en la gestión de la comunicación entre servicios.

### 3. ¿Qué es el Kernel?

El kernel es la parte central del sistema operativo que reside en memoria principal y proporciona las funciones más básicas, como la gestión de procesos, memoria y dispositivos de E/S. Opera en modo privilegiado y es responsable de garantizar la estabilidad y seguridad del sistema.

### 4. ¿Qué son los servicios?

Los servicios son funciones proporcionadas por el sistema operativo que permiten a los programas de aplicación realizar tareas específicas, como acceder a archivos, comunicarse con otros procesos o interactuar con dispositivos de hardware. Estos servicios se ofrecen a través de interfaces bien definidas.

### 5. ¿Qué son los controladores (drivers)?

Los controladores son módulos de software que permiten al sistema operativo comunicarse y controlar dispositivos de hardware específicos. Actúan como intermediarios, traduciendo las instrucciones del sistema operativo a comandos que el hardware puede entender y viceversa.

## **6. ¿Qué son las llamadas al sistema (system calls)?**

Las llamadas al sistema son interfaces proporcionadas por el sistema operativo que permiten a los programas de aplicación solicitar servicios específicos del kernel, como operaciones de entrada/salida, gestión de procesos o acceso a recursos del sistema. Estas llamadas proporcionan un mecanismo controlado y seguro para que las aplicaciones interactúen con el hardware y otros recursos críticos.

## **7. ¿Qué es un sistema multitarea (multitasking)?**

Un sistema multitarea es aquel que permite la ejecución concurrente de múltiples procesos o tareas. El sistema operativo gestiona la asignación de tiempo de CPU y otros recursos entre las tareas activas, proporcionando la ilusión de ejecución simultánea, incluso en sistemas con una sola CPU.

## **8. ¿Qué es un sistema multiusuario?**

Un sistema multiusuario es aquel que permite que múltiples usuarios accedan y utilicen los recursos del computador de manera simultánea. El sistema operativo gestiona y aísla las actividades de cada usuario, garantizando la seguridad, privacidad y equidad en el uso de los recursos.

## **9. ¿Qué es un sistema operativo de red y un sistema operativo distribuido?**

- **Sistema operativo de red:** Es un sistema que proporciona servicios para permitir que múltiples computadoras se comuniquen y compartan recursos a través de una red. Cada computadora mantiene su propio sistema operativo, pero se coordinan para ofrecer servicios de red, como acceso a archivos compartidos o impresoras.
- **Sistema operativo distribuido:** Es un sistema que gestiona un conjunto de computadoras independientes y las presenta al usuario como un único sistema coherente. El sistema operativo distribuye tareas y recursos entre las máquinas, proporcionando transparencia en la ubicación de los recursos y la ejecución de procesos.

## 1. ¿Qué es un proceso?

Un proceso es una abstracción de un programa en ejecución. Incluye el código del programa, su estado actual, y los recursos asignados, como registros, pila y espacio de direcciones. Es la unidad fundamental de trabajo en un sistema operativo.

## 2. ¿Qué es un subproceso?

Un subproceso es otro proceso con una menor jerarquía. **un subproceso es un proceso hijo**, creado por otro proceso mediante mecanismos del sistema operativo. Aunque puede estar jerárquicamente subordinado al proceso padre, su ejecución es autónoma y su entorno de ejecución es independiente. Este concepto debe distinguirse claramente del de **hilo**, el cual representa una unidad de ejecución más ligera que opera dentro de un proceso y comparte sus recursos.

## 3. ¿Para qué sirve el PCB? ¿Qué información lleva?

El Bloque de Control de Proceso (PCB) es una estructura de datos que almacena toda la información necesaria para gestionar y controlar un proceso. Incluye el estado del proceso, contador de programa, registros de CPU, información de planificación, información de gestión de memoria y recursos asignados.

## 4. ¿Qué eventos básicos pueden llevar a la creación de un proceso?

Los eventos que pueden llevar a la creación de un proceso incluyen:

- Inicio del sistema.
- Ejecución de una llamada al sistema por parte de un proceso en ejecución (por ejemplo, fork en UNIX).
- Solicitud del usuario para iniciar un nuevo programa.
- Inicio de un trabajo por parte del sistema operativo.

## 5. ¿Cuáles son los estados básicos de un proceso? Grafícelo como un diagrama de secuencia de estados.

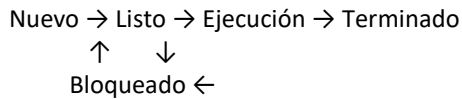
Los estados básicos de un proceso son:

- **Nuevo:** El proceso está siendo creado.



- **Listo:** El proceso está esperando ser asignado a un procesador.
- **Ejecución:** El proceso está siendo ejecutado por el procesador.
- **Bloqueado:** El proceso está esperando algún evento, como la finalización de una operación de E/S.
- **Terminado:** El proceso ha finalizado su ejecución.

El diagrama de transición de estados sería:



## 6. ¿Qué significa cada estado mencionado?

- **Nuevo:** El proceso está en fase de creación.
- **Listo:** El proceso está preparado para ejecutarse cuando se le asigne el procesador.
- **Ejecución:** El proceso está siendo ejecutado por el procesador.
- **Bloqueado:** El proceso no puede continuar hasta que ocurra un evento externo (por ejemplo, finalización de una operación de E/S).
- **Terminado:** El proceso ha completado su ejecución o ha sido finalizado por el sistema.

## 7. ¿Cuál es la diferencia entre proceso e hilo (thread)?

Un proceso es una instancia de un programa en ejecución con su propio espacio de direcciones y recursos. Un hilo es una unidad de ejecución dentro de un proceso que comparte el mismo espacio de direcciones y recursos del proceso principal, pero tiene su propio contador de programa y registros. Los hilos permiten una ejecución concurrente más eficiente dentro de un mismo proceso.

## 8. ¿Por qué es importante la programación multihilo?

La programación multihilo permite que múltiples hilos dentro de un proceso se ejecuten concurrentemente, lo que mejora la eficiencia y el rendimiento del sistema. Es especialmente útil en aplicaciones que realizan múltiples tareas simultáneamente, como servidores web o aplicaciones multimedia.

## 9. ¿Qué es el CPU scheduler (planificador de CPU)?

El planificador de CPU es el componente del sistema operativo responsable de seleccionar qué proceso en estado listo debe ser ejecutado por el procesador. Su objetivo es maximizar la utilización de la CPU y garantizar una asignación justa de tiempo de procesamiento entre los procesos.

## 10. ¿Cuál es la función del Dispatcher?

El Dispatcher es el módulo que da el control de la CPU al proceso seleccionado por el planificador. Realiza funciones como el cambio de contexto, el cambio al modo usuario y el salto a la ubicación adecuada en el programa del usuario para reanudar su ejecución.

### 11. ¿Qué diferencia hay entre el planificador a corto, mediano y largo plazo?

- **Planificador a largo plazo:** Determina qué procesos deben ser admitidos al sistema para su ejecución.
- **Planificador a mediano plazo:** Decide qué procesos deben ser suspendidos temporalmente para mejorar la utilización de la CPU y la memoria.
- **Planificador a corto plazo:** Selecciona qué proceso listo debe ser ejecutado a continuación por la CPU.

### 12. Describa el funcionamiento de los siguientes algoritmos para planificación de procesos:

- **FCFS (First-Come, First-Served):** Los procesos se ejecutan en el orden en que llegan.
- **Round Robin:** Cada proceso recibe un tiempo fijo de ejecución (quantum) y se alternan en orden circular.
- **SPN (Shortest Process Next):** Se ejecuta el proceso con el menor tiempo estimado de ejecución.
- **SRT (Shortest Remaining Time):** Se selecciona el proceso con el menor tiempo restante de ejecución.
- **HRRN (Highest Response Ratio Next):** Se prioriza el proceso con la mayor relación entre tiempo de espera y tiempo de servicio.
- **Feedback:** Los procesos se mueven entre diferentes colas de prioridad según su comportamiento y tiempo de ejecución.

### 13. ¿Qué significa que la planificación sea “preemptive” y qué significa que sea “non-preemptive”?

- **Preemptive:** El sistema operativo puede interrumpir un proceso en ejecución para asignar la CPU a otro proceso.
- **Non-preemptive:** Un proceso en ejecución no puede ser interrumpido hasta que finalice o entre en estado de espera.

### 14. ¿Qué sucede si se define un quantum demasiado pequeño en un algoritmo Round Robin?

Si el quantum es demasiado pequeño, se producen cambios de contexto frecuentes, lo que aumenta la sobrecarga del sistema y puede disminuir la eficiencia.

### **15. ¿Qué sucede si se define un quantum demasiado grande en un algoritmo Round Robin?**

Si el quantum es demasiado grande, el sistema se comporta de manera similar a FCFS, lo que puede aumentar el tiempo de respuesta para procesos cortos.

### **16. Un algoritmo preemptive ¿Qué dificultad puede traer?**

Los algoritmos preemptive pueden aumentar la complejidad del sistema operativo y la sobrecarga debido a los frecuentes cambios de contexto. También pueden causar problemas de sincronización si no se gestionan adecuadamente.

### **17. ¿Por qué se organiza la planificación en colas?**

La planificación en colas permite gestionar y priorizar eficientemente los procesos según su estado y características, facilitando la selección del próximo proceso a ejecutar y mejorando la utilización de los recursos del sistema.

### **18. ¿Qué situaciones llevan a un deadlock (interbloqueo o abrazo mortal)?**

Un deadlock ocurre cuando un conjunto de procesos queda bloqueado esperando recursos que están siendo retenidos por otros procesos del mismo conjunto. Las condiciones necesarias para un deadlock son: exclusión mutua, retención y espera, no apropiación y espera circular.

### **19. ¿A qué se denomina Sección Crítica y qué problemas se aparejan con ella?**

La Sección Crítica es una parte del código donde un proceso accede a recursos compartidos. Si no se gestiona adecuadamente, puede llevar a condiciones de carrera y resultados inconsistentes debido al acceso concurrente de múltiples procesos o hilos.

### **20. ¿Qué mecanismos existen para la protección de la Sección Crítica?**

Existen varios mecanismos para proteger la Sección Crítica, como:

- **Semáforos:** Variables especiales utilizadas para controlar el acceso a recursos compartidos.
- **Monitores:** Estructuras que permiten la sincronización de procesos mediante procedimientos y variables compartidas.
- **Mutexes:** Objetos que permiten la exclusión mutua, asegurando que solo un hilo acceda a la Sección Crítica a la vez.