

1. Algoritmos en sección crítica, semáforos y procesos con multihilos:

- Sección Crítica: Un área del código que debe ser ejecutada de manera exclusiva por un proceso a la vez. Pueden implementarse con técnicas como semáforos, mutex o variables de condición.
- Semáforos: Mecanismo de sincronización que permite el acceso a recursos compartidos. Pueden ser binarios (0 o 1) o contar el número de procesos que pueden acceder.
- Procesos con Multihilos: La concurrencia se logra mediante hilos que comparten recursos y ejecutan tareas independientes. Se utilizan técnicas como semáforos y mutex para garantizar la sincronización.

[Microsoft learning](#)

[Wikipedia](#)

[GeeksForGeeks](#)

2. Compartición de memoria en el sistema operativo: La compartición de memoria permite que varios procesos accedan a la misma área de memoria. Un ejemplo en código en C sería el uso de la función `shmget` y `shmat` para crear y adjuntar una región de memoria compartida.

```
c #include <sys/ipc.h> #include <sys/shm.h>
```

```
int shmid = shmget(key, size, IPC_CREAT | 0666); char shared_memory = (char) shmat(shmid, NULL, 0);
```

[Scales topics](#)

3. Reubicación por hardware en la memoria: La reubicación por hardware implica que el hardware se encarga de ajustar las direcciones de memoria en tiempo de ejecución. Un ejemplo en código sería el uso de registros de base y límite en una arquitectura que lo soporte.

```
assembly MOV AX, 1234 ; Valor base MOV DS, AX ; Establecer registro de segmento de datos
```

[StackOverflow](#)

[Wikipedia](#)

4. Reubicación por software en la memoria: En este caso, el sistema operativo se encarga de ajustar las direcciones de memoria. Un ejemplo en código puede ser el uso de registros relativos.

```
c int base_address = 0x1000; // Dirección base int offset = 0x0100; // Desplazamiento int result = base_address + offset;
```

[Wikipedia](#)

5. Zonas compartidas en memoria: Son áreas de memoria que pueden ser accedidas por varios procesos. Por ejemplo, segmentos compartidos de memoria o memoria compartida entre procesos.

[StackOverflow](#)

[Boost Org](#)

6. Mapas de procesos: Son estructuras que describen la disposición de la memoria virtual de un proceso. Tipos incluyen el mapa de código, datos, montajes, y pila. Sirven para entender cómo está organizada la memoria de un proceso.

[Asana](#)

7. Biblioteca de objetos en la memoria: Se refiere a las bibliotecas compartidas (shared libraries) que contienen funciones y datos que pueden ser compartidos entre varios programas en ejecución.

[CppReference](#)

8. Bibliotecas dinámicas en el sistema operativo: Son bibliotecas que se cargan en tiempo de ejecución. Los programas pueden enlazarse dinámicamente a estas bibliotecas.

[GeeksForGeeks](#)

9. Variables globales y locales en la memoria: Las variables globales existen durante toda la ejecución del programa, mientras que las locales solo existen dentro de su ámbito de definición (como una función).

[Online tutorials point](#)

10. Creación de mapa de memoria inicial a partir de ejecutables: Cuando se ejecuta un programa, el sistema operativo crea un mapa de memoria inicial, reservando espacio para el código, datos, pila, y otros segmentos.

[Universidad Carlos III de madrid - Gestión de memoria en Servidores de

Información Multimedia](<https://www.it.uc3m.es/labsimitis/sesiones/teoria/gestionmemoria.pdf>)

11. Procesos después de generar nuevas regiones en los mapas de memoria: Después de generar nuevas regiones en los mapas de memoria, el sistema operativo realiza la asignación física de memoria y actualiza los registros de la tabla de páginas.

[CodeCademy](#)

12. Archivos proyectados en memoria: Permiten que el contenido de un archivo sea tratado como si estuviera en memoria principal. Un ejemplo es el mapeo de un archivo en memoria usando `mmap` en C.

[Win32 ProjFS](#)

[Win32 ProjFS](#)

[Win32 ProjFS](#)

13. Servicio de gestión de memoria: Es un componente del sistema operativo encargado de gestionar la asignación y liberación de memoria a los procesos.

[GeeksForGeeks](#)

14. Proyección en memoria en el uso de bibliotecas dinámicas: Las bibliotecas dinámicas son proyectadas en memoria durante la ejecución del programa cuando se cargan dinámicamente.

[Win32 ProjFS](#)

[Win32 ProjFS](#)

[Win32 ProjFS](#)

15. Proyección POSIX: En POSIX, `mmap` se utiliza para proyectar archivos en memoria y manipularlos como regiones de memoria.

[Wikipedia](#)

[Baeldung](#)

16. Paginación: Es una técnica de gestión de memoria que divide la memoria en páginas. La tabla de páginas mapea direcciones virtuales a direcciones físicas.

[GeeksForGeeks](#)

17. Paginación por zonas: Divide la memoria en zonas lógicas (código, datos, pila) y utiliza tablas de páginas diferentes para cada zona.

[GeeksForGeeks](#)

18. Paginación con gestión de swap: Permite que páginas no usadas se almacenen en el disco para liberar espacio en memoria RAM.

[GeeksForGeeks](#)

19. Paginación con pre asignación de swap: Asigna espacio en el disco antes de que sea necesario, anticipando la necesidad de usar el área de swap.

[GeeksForGeeks](#)

20. Paginación sin pre asignación de código: Se refiere a la paginación donde las páginas de código no se cargan en memoria hasta que son necesarias.

[GeeksForGeeks](#)

21. Paginación por Mecanismo de Traducción: Emplea un mecanismo (generalmente una tabla de páginas) para traducir direcciones virtuales a direcciones físicas.

[GeeksForGeeks](#)

22. Tipos de modelos de procesos:

- Modelo de Proceso Cooperativo: Los procesos cooperan para lograr un objetivo.
- Modelo de Proceso Competitivo: Los procesos compiten por recursos. Ejemplo en C:

```
// Cooperativo pthread_mutex_lock(&mutex); // sección crítica  
pthread_mutex_unlock(&mutex);
```

```
// Competitivo int x = 0; // proceso 1 x = x + 1; // proceso 2 x = x * 2;
```

[GNU Org](#)

23. Jerarquía, Bloque de Control de Proceso (BCP): La jerarquía BCP es una estructura que contiene información sobre un proceso, incluyendo su estado, registros, identificador, prioridad, etc.

[Ginzo](#)

24. Sistemas embebidos: Son sistemas dedicados a tareas específicas. Ejemplos incluyen sistemas en automóviles, electrodomésticos, etc. Pueden utilizar sistemas operativos en tiempo real o sistemas operativos adaptados a recursos limitados.

[Wikipedia](#)

25. ¿Qué son los sistemas operativos en tiempo real y cuáles son sus características de funcionamiento?

Los sistemas operativos en tiempo real están diseñados para responder a eventos en un tiempo determinado, conocido como tiempo de respuesta máximo. Se utilizan en aplicaciones donde la predicción y cumplimiento de plazos son críticos. Algunas características clave incluyen el determinismo, con plazos duros o blandos, la asignación de prioridades a las tareas y un bajo tiempo de latencia para respuestas rápidas.

[Azure](#)

26. ¿Qué son las interrupciones y cuáles son sus funciones en el sistema operativo? Las interrupciones son señales que detienen la ejecución normal de un procesador para manejar eventos específicos. Funcionan para manejar eventos externos como pulsaciones de teclas o errores, mejorar la eficiencia permitiendo que la CPU realice otras tareas mientras espera eventos, y gestionar la comunicación entre hardware y software.

[GeeksForGeeks](#)

[Wikipedia](#)

27. ¿Cuál es la función de cada elemento de la estructura del software? La estructura del software incluye componentes como sistemas operativos, controladores y aplicaciones. El sistema operativo administra recursos y ofrece servicios a las aplicaciones. Los controladores facilitan la comunicación entre hardware y software, mientras que las aplicaciones son programas que realizan tareas específicas.

[GeeksForGeeks](#)

28. ¿Cuál es el diagrama de la estructura del software de entrada y salida? El diagrama de la estructura de entrada y salida muestra cómo los datos entran y salen del sistema. Incluye el sistema operativo, controladores y dispositivos de entrada/salida. La interacción entre estos elementos garantiza un manejo eficiente y seguro de la entrada y salida de datos.

[W3Schools](#)

29. ¿Cuál es el diagrama del modo de usuario y kernel y cómo funciona? El diagrama del modo de usuario y kernel ilustra la transición entre los modos de operación. El kernel tiene acceso a recursos críticos, mientras que el usuario opera en un entorno más restringido. Las transiciones entre estos modos son controladas por instrucciones privilegiadas, y el modo kernel garantiza un control total sobre el hardware.

[Baeldung](#)

30. ¿Cómo funciona el DMA en la estructura de procesos? El Acceso Directo a Memoria (DMA) es un mecanismo que permite a dispositivos transferir datos directamente a la memoria sin intervención de la CPU. En la estructura de procesos, el DMA mejora la eficiencia al reducir la carga de la CPU en operaciones de transferencia de datos, permitiéndole realizar otras tareas mientras el DMA se encarga de la transferencia.

[GeeksForGeeks](#)

31. ¿Qué es el caching y cómo funciona con el DMA y procesos? La caché es una memoria rápida que almacena datos utilizados con frecuencia para acelerar su acceso. En el contexto del DMA y procesos, la caché puede mejorar el rendimiento al reducir los tiempos de acceso a la memoria principal. La utilización eficiente de la caché puede minimizar la espera y mejorar la velocidad general del sistema.

[GeeksForGeeks](#)

32. ¿Qué es el system call y cuál es su diagrama? Una System Call es una interfaz que permite a las aplicaciones solicitar servicios del kernel del sistema operativo. El diagrama de una System Call muestra cómo una aplicación, en modo de usuario, realiza una llamada al sistema para solicitar un servicio. La transición al modo kernel permite al kernel realizar la tarea solicitada y devolver los resultados a la aplicación.

[PhoenixNap](#)

33. ¿Cuál es la estructura de microkernel y cómo funciona? Un microkernel es un diseño de sistema operativo que implementa solo funciones esenciales en el núcleo, mientras que servicios adicionales se ejecutan como procesos en espacio de usuario. Este enfoque reduce la complejidad y mejora la flexibilidad del sistema operativo al permitir la adición y eliminación de servicios sin afectar el núcleo.

[GeeksForGeeks](#)

34. ¿Qué es el sistema de lotes y para qué sirve? El sistema de lotes ejecuta tareas sin intervención directa del usuario. Se utiliza para procesar grandes volúmenes de datos o ejecutar trabajos programados en lotes. Los usuarios envían trabajos al sistema de lotes, que los procesa de manera secuencial sin necesidad de interacción continua.

[Tecnofaq](#)

35. ¿Cuáles son los estados de procesos? Los estados típicos de procesos incluyen "Listo", "En ejecución" y "Bloqueado". Un proceso está "Listo" cuando está preparado para ejecutarse, "En ejecución" cuando está siendo procesado por la CPU y "Bloqueado" cuando está temporalmente detenido, por ejemplo, esperando entrada/salida.

[BarcelonaGeeks](#)

36. ¿Qué es y cómo funciona el algoritmo de Lamport? El algoritmo de Lamport se utiliza para sincronizar procesos distribuidos. Emplea relojes lógicos para ordenar eventos en diferentes nodos. Cada proceso mantiene un reloj lógico que refleja el orden de los eventos. Los relojes lógicos permiten la coordinación y la resolución de conflictos en sistemas distribuidos.

[GeeksForGeeks](#)

37. ¿Qué es y cómo funciona el algoritmo de Peterson? El algoritmo de Peterson es un método para lograr la exclusión mutua en sistemas con acceso compartido. Utiliza variables compartidas y señales para garantizar que solo un proceso acceda a un recurso crítico a la vez, evitando la interferencia entre procesos concurrentes.

[GeeksForGeeks](#)

38. ¿Para qué sirve sincronizar procesos con un algoritmo de "espera no activa"? La sincronización de procesos con un algoritmo de "espera no activa" se utiliza para evitar el uso intensivo de recursos cuando no es necesario. Los procesos en espera no activa evitan la comprobación constante de condiciones, reduciendo la carga de la CPU y mejorando la eficiencia en situaciones de espera prolongada.

[Wikipedia](#)

39. ¿Qué es y cómo funciona la "exclusión mutua"? La exclusión mutua garantiza que solo un proceso tenga acceso a un recurso crítico en un momento dado. Métodos como semáforos y mutex se utilizan para implementar la exclusión mutua. Esto evita conflictos y asegura que las operaciones críticas se realicen de manera ordenada y sin interferencias.

[Wikipedia](#)

40. ¿Cómo funcionan las Secciones Críticas Anidadas y la Sección Crítica? Las Secciones Críticas Anidadas ocurren cuando un proceso intenta ingresar a una sección crítica mientras ya está dentro de otra. Puede llevar a problemas de bloqueo si no se maneja adecuadamente. La Sección Crítica es un área de código que debe ejecutarse de manera atómica, evitando interferencias entre procesos concurrentes.

[TutorialsPoint](#)

41. ¿Qué es la Orden de Adquisición de Bloqueos y para qué sirve? La Orden de Adquisición de Bloqueos especifica un orden particular en el que los procesos deben adquirir bloqueos para evitar el riesgo de bloqueo mutuo. Esto previene situaciones en las que dos procesos esperan indefinidamente el bloqueo del otro, evitando problemas de inanición y deadlock.

[GeeksForGeeks](#)

42. ¿Qué son Los Semáforos Binarios y cómo funcionan? Los Semáforos Binarios son un tipo de semáforo con dos estados: 0 y 1. Se utilizan para controlar el acceso a recursos compartidos, permitiendo o bloqueando el acceso según el estado del semáforo. Un semáforo binario puede ser utilizado para implementar la exclusión mutua y la sincronización entre procesos.

[Wikipedia](#)

43. ¿Cuál es la estructura de la jerarquía de la memoria y su función? La jerarquía de la memoria incluye registros, caché, memoria principal y almacenamiento secundario. Su función es proporcionar diferentes niveles de almacenamiento con diferentes velocidades y capacidades para optimizar el rendimiento del sistema.

[GeeksForGeeks](#)

44. ¿Qué son las tablas de procesos y para qué sirven? Las tablas de procesos mantienen información sobre los procesos en ejecución. Contienen detalles como el estado del proceso, el contador de programa y la información de recursos asignados. Sirven para gestionar y supervisar el estado y la ejecución de los procesos en el sistema operativo.

"Operating System Concepts" de Abraham Silberschatz, Peter B. Galvin, Greg Gagne

45. ¿Qué es y cómo funciona el IPC de procesos? La Comunicación entre Procesos (IPC) permite que los procesos compartan información y se comuniquen entre sí. Métodos como colas de mensajes, semáforos y tuberías facilitan la transferencia de datos entre procesos, permitiendo la cooperación y la sincronización en sistemas operativos multitarea.

"Advanced Programming in the UNIX Environment" de W. Richard Stevens

46. ¿Para qué sirven las técnicas de sincronización de procesos? Las técnicas de sincronización de procesos se utilizan para coordinar y controlar el acceso a recursos compartidos en entornos multitarea. Evitan condiciones de carrera y garantizan la consistencia de los datos al sincronizar las operaciones entre procesos concurrentes.

"Operating System Concepts" de Abraham Silberschatz, Peter B. Galvin, Greg Gagne

47. ¿Qué son, ¿cuáles existen y su función de algoritmos de planificación en los sistemas operativos? Los algoritmos de planificación en sistemas operativos determinan el orden en que los procesos se ejecutan. Algunos algoritmos incluyen FCFS (Primero en llegar, primero en ser servido), SJF (Planificación de trabajo más corto primero) y Round Robin (Asignación de

tiempo en rodajas). Estos algoritmos optimizan el rendimiento del sistema al asignar recursos de manera eficiente.

[Tutorials](#)

48. ¿Qué es el Bootstrap? El Bootstrap es el proceso de carga inicial del sistema operativo cuando se inicia la computadora. Durante el Bootstrap, se carga el kernel del sistema operativo en la memoria y se inicializan los componentes esenciales del sistema para que esté listo para su funcionamiento.

"Modern Operating Systems" de Andrew S. Tanenbaum

49. ¿Cuáles son las principales características de una interrupción de hardware, una excepción y una trampa?

- Interrupción de hardware: Generada por eventos externos como temporizadores, pulsaciones de teclas, etc.
- Excepción: Resulta de errores o condiciones anómalas durante la ejecución del programa.
- Trampa: Intencionalmente insertada para interrumpir la ejecución normal del programa.

[GeeksForGeeks](#)

50. ¿Cuáles son los criterios de planificación y para qué sirven? Los criterios de planificación incluyen factores como la prioridad, el tiempo de espera, la equidad y la utilización de CPU. Sirven para determinar el orden en que los procesos se ejecutan, optimizando el rendimiento del sistema operativo y satisfaciendo los requisitos de los usuarios y las aplicaciones.

"Operating System Concepts" de Abraham Silberschatz, Peter Baer Galvin y Greg Gagne.

51. Modelos Multihilos:

- Definición: Los hilos (threads) son subprocesos ligeros que comparten recursos de un mismo proceso.
- Tipos: "Operating System Concepts" de Abraham Silberschatz, Peter Baer Galvin y Greg Gagne.
 - Multihilo a nivel de usuario (ULP): Administrados por la aplicación, sin intervención del sistema operativo.
 - Multihilo a nivel de kernel (KLP): Administrados por el sistema operativo.
- Clasificación:
 - Hilos de Usuario: Creados y gestionados por la aplicación.
 - Hilos del Kernel: Administrados por el sistema operativo.

"Modern Operating Systems" de Andrew S. Tanenbaum

52. Tuberías y Memoria Compartida:

- Tuberías: Canal de comunicación entre procesos, permitiendo la transferencia de datos de un extremo a otro.
- Memoria Compartida: Región de memoria compartida entre procesos que permite la comunicación directa.

"UNIX Network Programming" de W. Richard Stevens.

53. Principios de Sincronización:

- Condición de Carrera: Competencia por recursos compartidos.
- Sección Crítica: Región del código que debe ejecutarse de forma exclusiva.
- Exclusión Mutua: Garantiza que solo un proceso pueda ejecutar una sección crítica a la vez.
- Atomicidad: Operación que se realiza de forma ininterrumpida.

- Abrazo Mortal: Situación donde dos o más procesos se bloquean esperando a que el otro libere un recurso.

"Modern Operating Systems" de Andrew S. Tanenbaum. "Operating System Concepts" de Abraham Silberschatz, Peter B. Galvin y Greg Gagne.

54. Métodos de Asignación de Espacio:

- Compactación: Reorganiza la memoria para consolidar huecos libres.
- Partición Fija: Divide la memoria en particiones fijas.
- Partición Variable: Divide la memoria en particiones de tamaño variable.
- Paginación: Divide la memoria en páginas fijas.
- Segmentación: Divide la memoria en segmentos de tamaño variable.
- Paginación Segmentada: Combinación de paginación y segmentación.

Tanenbaum, A. S., & Bos, H. (2014). "Sistemas Operativos Modernos" (Capítulos 3 y 4). Stallings, W. (2014). "Operating Systems: Internals and Design Principles" (Capítulos 8 y 9).