

Respuestas Orientadoras e incompletas (en algunos casos) a las preguntas de revisión. Favor tomarlas como orientadoras.

Clase 2 – Revisión

1. ¿Cómo funciona una pila o stack? ¿Cuál es su punto de acceso, su límite y su base? ¿Es una lista de tipo LIFO o FIFO?

Una pila es un conjunto ordenado de elementos, en donde el último elemento en entrar al conjunto es el primero en salir, se puede implementar como una lista de tipo LIFO (last-in, first-out). Su punto de acceso es a través de la cima (desde allí se pueden agregar o quitar elementos)... El uso habitual es:

push a
push b
push c

...

...

pop c
pop b
pop a

2. ¿Qué utilidad puede tener la utilización de una estructura de tipo pila o stack? ¿Qué implicaciones en cuanto a la programación tiene este tipo de estructura?

Tiene gran aplicación en la ejecución de procesos, por ejemplo, en un sistema multiprogramado, la cpu puede estar ejecutando el proceso A, supongamos que éste realiza una operación de E-S, la cpu puede "apilar" toda la información del proceso A, interrumpirlo, poner en ejecución el proceso B y si luego que quiere retomar la ejecución del proceso A en el punto en el que había quedado, se puede "desapilar" la información del proceso A. Las implicancias en cuanto a la programación estan indicadas en el punto anterior, en cuanto al orden de "apilamiento" que hay que respetar, si apilo los datos de los registros a,b,c debo desapilarlos en el orden inverso (c,b,a).

3. ¿Qué relación hay entre una pila o stack y las operaciones PUSH y POP?

Push -> apilar, poner

Pop -> desapilar, sacar

4. ¿Cuáles son los objetivos de un SO?

1-SO como interfaz de usuario...

2-Eficiencia (uso eficiente de los recursos del computador)...

3-Evolución (permitir agregar nuevas funciones sin afectar a los servicios que brinda)...

5. Una vez que el SO está en funcionamiento, ¿El núcleo o kernel siempre permanece cargado en memoria? O bien el mismo se carga y descarga de la memoria según lo requiera la administración de memoria del SO.

Siempre permanece cargado en memoria.

6. ¿Cuáles eran los principales problemas del procesamiento en Serie de los primeros SO's?

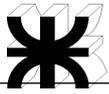
-Planificación: uso de computadores costosos como monousuarios, monotarea.

-Tiempo de Preparación: cargar compilador, cargar programa, guardar programa compilado, cargar programa compilado, montar y desmontar cintas, etc.

7. En el procesamiento en Serie, ¿había algún tipo de aislación (isolation) entre el usuario y el hardware utilizado?

No.

8. En el procesamiento batch o por lotes, ¿había algún tipo de aislación (isolation) entre el usuario y el hardware utilizado?



Si, el monitor.

9. ¿De qué manera el usuario podía interactuar con el Monitor?

A través del conjunto de instrucciones JCL (job control language) que se acompañaba con cada trabajo a ejecutar.

10. En un sistema de procesamiento batch o por lotes, ¿Qué sucede cuando un programa del usuario ejecuta una instrucción privilegiada (asumiendo que el hardware tiene la facilidad de contar con este tipo de instrucciones)? ¿Cuál cree Ud. que es la razón de ello?

Si un programa de usuario pretende ejecutar una instrucción privilegiada se producirá un error, puesto que este tipo de instrucciones no esta permitido para un programa de usuario. Solo el monitor puede ejecutar instrucciones privilegiadas puesto que éstas tienen mayor prioridad que las instrucciones del usuario, si los programas de usuario pudieran ejecutar estas instrucciones habría -al menos- dos problemas: 1. si un programa de usuario acaparara todo el tiempo de cpu ejecutando instrucciones privilegiadas (de mayor prioridad), el monitor no podría tomar el control. 2. se perdería la aislación / abstracción del hardware que provee el monitor, la interacción con el hardware debe hacerse a través del monitor y no directamente desde los programas de usuario.

11. Las instrucciones privilegiadas incluyen también operaciones de I/O, entonces, ¿Cómo es posible que un programa de usuario pueda realizar operaciones de I/O?

Porque las hace a través del monitor y no directamente.

12. ¿Qué áreas de memoria están disponibles para un programa de usuario cuando éste se ejecuta en “modo usuario” (user mode)?

Solo puede acceder a su propia area de datos y aquellas a las que se le haya otorgado acceso.

No puede acceder a areas de memoria del monitor o de otro proceso.

13. ¿Qué áreas de memoria están disponibles para el monitor teniendo en cuenta que éste se ejecuta en “modo núcleo” (kernel mode)?

El monitor puede acceder a cualquier area de memoria.

14. En un sistema de tiempo compartido o time sharing, ¿el sistema de procesamiento batch o por lotes es reemplazado por la terminal? O bien son sistemas en donde pueden convivir tanto el procesamiento batch como procesos interactivos.

Son sistemas en donde pueden convivir tanto el procesamiento batch como procesos interactivos.

La terminal es solo un proceso más.