

## **EXAMEN DE FUNDAMENTOS DEL SOFTWARE: TEMAS 1 Y 2**

**1.- Suponga una máquina (imaginaria) que direcciona por palabras de 32 bits. Su memoria principal está organizada en 256 marcos de página y cada marco es de 1 KB. Por otra parte, las instrucciones máquina usan 8 bits para el código de instrucción. En este supuesto:**

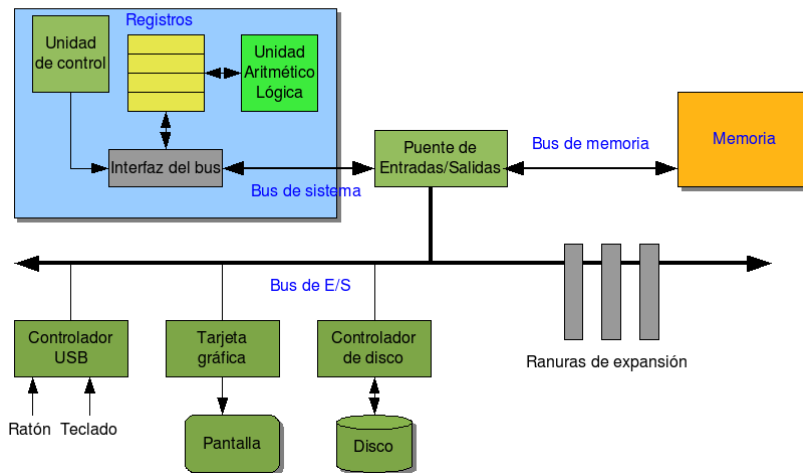
- \* a) ¿Cuántos bits serían necesarios para un acceso a memoria?**
- \* b) ¿Cuántas instrucciones máquina puede tener implementadas como máximo ese procesador?**
- \* c) ¿Qué tipos de instrucciones máquina debería incluir al menos el procesador para ser considerado de propósito general (poder resolver cualquier tipo de problema)?**

**2.- Diferencias entre lenguaje máquina, ensamblador y lenguaje de alto nivel (de tercera generación).**

El lenguaje máquina es el lenguaje cuyas instrucciones interpretan los circuitos eléctricos de la unidad de control. El ensamblador, por el contrario, es el encargado de traducir de lenguaje ensamblador a lenguaje máquina. Y el lenguaje de alto nivel, es el encargado de mostrar distintos algoritmos de modo que tal usuario pueda leerlo, reorganizarlo e interpretarlo, distanciándose del lenguaje propio de los ordenadores.

**3.- a) Enumere los componentes principales de una arquitectura básica para un sistema de computación.**

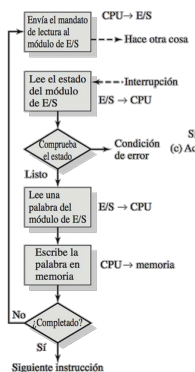
Procesador



**b) Describa la funcionalidad de los diversos tipos de registros del procesador.**

- Registros visibles para el programador.
- Registros de control y estado:
  - Contador de programa (PC).
  - Puntero de pila (SP).
  - Registro de instrucción (IR).
  - Registro de estado (bits informativos).

**4.- a) Ciclo de instrucción de una E/S dirigida por interrupciones.**



**b) Diferencias entre monoprogramación, multiprogramación y tiempo compartido.**

b) Los sistemas en los que existe un único programase denomina monoprogramación. El procesador, ejecuta durante un cierto tiempo hasta que llega a una instrucción E/S, entonces debe esperar a que esta concluya para continuar. Sin embargo, esta ineficiencia se puede evitar ya que

hay memoria suficiente para albergar al S.O y varios programas usuario de modo que cuando un proceso tiene que esperar por la E/S se asigna el procesador a otro proceso. Esto es lo que se conoce como multiprogramación. En los sistemas de tiempo compartido, se utiliza la multiprogramación para gestionar múltiples trabajos interactivos. Compartiendo el tiempo de procesador entre múltiples usuarios que acceden simultáneamente a través de terminales, el S.O se encarga de entrelazar la ejecución de cada programa de usuario.

#### **5.- a) Diferencias entre el modo supervisor y el modo usuario en la ejecución de instrucciones máquina.**

En el modo supervisor (o modo kernel) el programa, (S.O) que se ejecuta tiene acceso a todos los recursos de la máquina tanto software como hardware mientras que en el modo usuario, el programa que se ejecuta en este modo solo tiene acceso a un subconjunto de registros del procesador, un subconjunto del repertorio de instrucciones máquina y a un área de la memoria.

#### **En el contexto de los modos de ejecución de instrucciones:**

\* **b) ¿qué se entiende por "trampa"?** Podríamos decir que una trampa es una interrupción generada por el SOFTWARE debido a un error (por ejemplo, seg fault), o a una solicitud específica de un programa de usuario que necesita del S.O. Cuando ocurren el hardware transfiere el control al S.O.

\* **c) ¿qué acciones deben llevarse a cabo en una "trampa"?** La instrucción TRAP genera la interrupción. El procesador lo acepta y se ejecuta la rutina de interrupción determinando el punto de acceso al servicio solicitado. Se llama al servicio y se ejecuta el correspondiente código. Finalmente se retorna a la rutina genérica, restituyendo los registros y volviendo a la interrupción siguiente al TRAP.

#### **6.- Concepto de:**

\* **a) Programa / Proceso / Hebra.** Un programa es un conjunto de instrucciones almacenadas en memoria. Un proceso es la unidad de actividad caracterizada por un hilo secuencias de ejecución, un estado actual y un conjunto de recursos del sistema asociados. Está formado por un programa ejecutable, una instancia de un programa ejecutándose y para ejecutarse necesita datos del contexto de ejecución y del programa. Además, permite al S.O. controlar la asignación de los recursos necesarios para la ejecución de programas y controlar la ejecución del programa asociado al proceso de forma intercalada con otros programas. Sin embargo, el concepto de hebra separa estas dos características. La tarea se encarga de soportar todos los recursos necesarios (incluida la memoria) y cada una de las hebras permite la ejecución del programa de forma "independiente" del resto de hebras.

\* **b) ¿Qué información debe contener al menos el Bloque de Control de Proceso (PCB)?** Debe contener al menos la información sobre el identificador de proceso, contexto de ejecución, memoria donde reside el programa y sus datos, información relacionada con recursos del sistema y del estado del proceso.

#### **7.- Describa de forma breve, pero completa, el Modelo de Cinco Estados de los procesos.**

El modelo de 5 estados trata de representar las actividades que el S.O. lleva a cabo sobre los procesos: creación, terminación, y multiprogramación. Para ello hace uso de 5 estados:

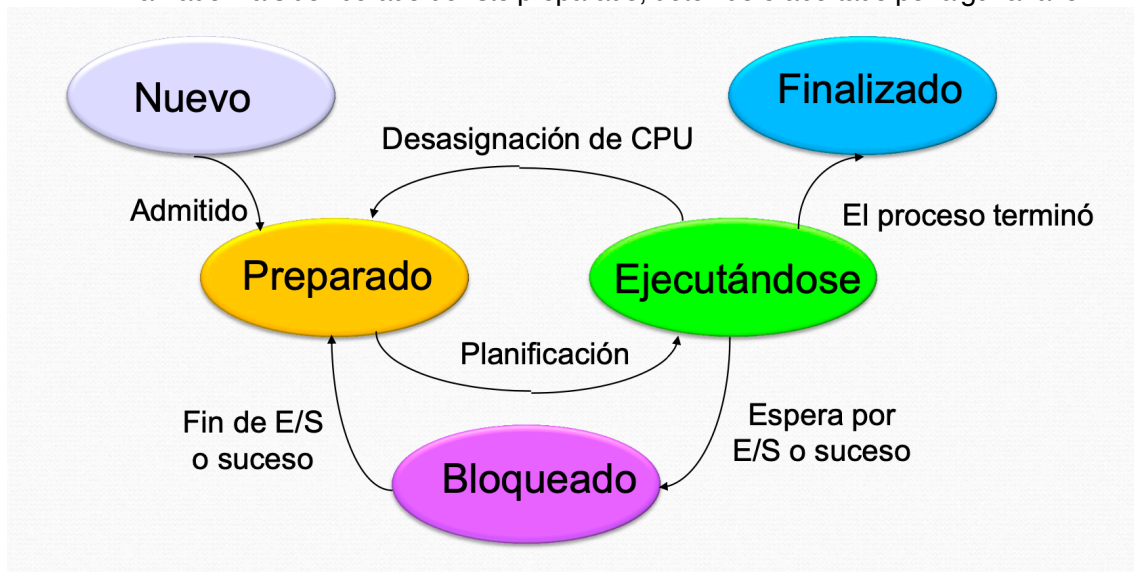
Nuevo: se acaba de crear y aún no ha sido admitido en listo por el S.O.

Preparado: listo para ejecutarse.

Ejecutando: en ejecución.

Bloqueado: no puede ejecutar nada hasta que se cumpla un evento.

Finalizado: ha sido liberado de listo/preparado, detenido o abortado por alguna razón.



#### 8.-Pasos seguidos durante un cambio de contexto de ejecución.

- 1) Salvar los registros del procesador en el PCB del proceso que actualmente está en estado "Ejecutándose".
- 2) Actualizar el campo estado del proceso al nuevo estado al que pasa e insertar el PCB en la cola correspondiente.
- 3) Seleccionar un nuevo proceso del conjunto de los que se encuentran en estado "Preparado".
- 4) Actualizar el estado del proceso seleccionado a "Ejecutándose" y sacarlo de la cola de preparados.
- 5) Cargar los registros del procesador con información de los registros almacenado en el PCB del proceso seleccionado.

#### 9.- a) ¿Qué es la reubicación? b) ¿Qué es la reubicación estática? c) ¿Y la dinámica?

La REUBICACIÓN es la capacidad de cargar y ejecutar un programa en un lugar arbitrario de la memoria para maximizar la utilización de la CPU. En la REUBICACIÓN ESTÁTICA, el compilador genera direcciones lógicas de 0 a M y la decisión de donde ubicar el programa en memoria principal se realiza en tiempo de carga. El cargador añade la dirección base de carga a todas las referencias relativas a memoria del programa. En la REUBICACIÓN DINÁMICA, el compilador genera direcciones lógicas de 0 a M. La traducción de direcciones lógicas a físicas se realiza en tiempo de ejecución luego el programa está cargado con referencias relativas. Requiere el apoyo del hardware.

#### 10.- Describa los siguientes conceptos:

##### \* a) Paginación, marco, página de memoria.

Como solución a la fragmentación de memoria aparecen la paginación y la segmentación. La paginación trata de trocear el espacio lógico en páginas que son elementos de longitud fija. La memoria se divide en porciones de tamaño fijo relativamente pequeños denominados marcos de página. El tamaño es potencia de dos, de 512 B a 8KB. El espacio lógico de un proceso se divide conceptualmente en bloques del mismo tamaño, denominados páginas. Los marcos de página contendrán páginas de los procesos.

**\* b) Segmentación, segmento de memoria.**

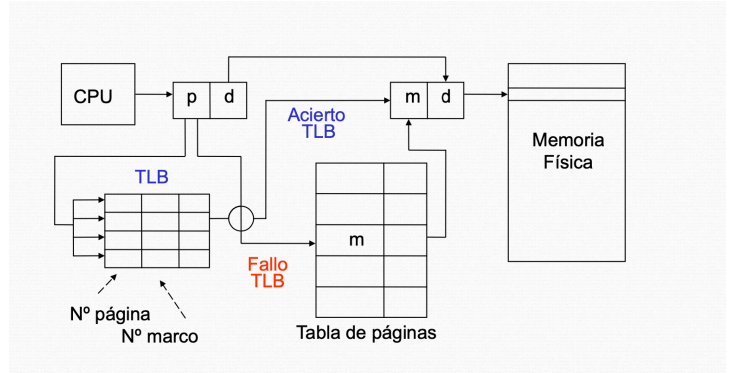
Esta otra solución a la fragmentación de memoria, trata de trocear el espacio lógico pero en segmentos que son elementos de longitud variable. Es similar al particionamiento dinámico pero con una diferencia, el programa puede ocupar más de una partición y no necesitan ser contiguas. La paginación es invisible al programador mientras que la segmentación es visible y de mayor utilidad para organizar programas y datos.

**\*c) Esquema de traducción de direcciones**

El problema de los dos accesos a memoria se resuelve con una caché hardware de consulta rápida denominada búfer de traducción adelantada o TLB (Translation Look-aside Buffer). El TLB se implementa como un conjunto de registros asociativos que permiten una búsqueda en paralelo. De esta forma, para traducir una dirección:

1 Si existe ya en el registro asociativo, obtenemos el marco.

2 Si no, la buscamos en la tabla de páginas y se actualiza el TLB con esta nueva entrada.



**11.- Suponga una máquina de memoria de 2GB, marcos de página de 1KB y un proceso que ocupa 6 páginas. Si la tabla de páginas incluye un bit de protección para cada entrada, ¿cuántos bytes ocuparía dicha tabla?**

12.- Considere la siguiente tabla de páginas para un proceso en un sistema con gestión de memoria paginada. Si el tamaño de página es de 2048 bytes, ¿qué dirección física corresponde a cada una de las direcciones lógicas que se dan a continuación? En caso de no poder realizar la traducción, explique por qué no se puede:

Número Página	Número Marco	Página
0	3	
1	13	
2	7	
3	8	
4	14	

\* a) 2048

\* b) 37

\* c) 13055

\* d) 10128