

## Unidades y funciones matemáticas usadas en el texto

Las unidades básicas de almacenamiento u organización de la información son el *bit* (símbolo b) Y el *byte* (símbolo B). Un bit es un dígito binario, es decir, un 0 o un 1, que se utiliza para representar una pareja de posibles estados, por ejemplo: encendido y apagado, presente y ausente, etc. Un byte son ocho bits contiguos.

Prefijo Símbolo Valor Sistema de unidades

Prefijo	Símbolo	Valor	Sistema de unidades
kibi	Ki	$2^{10}$	IEEE 1541
kilo	K	$10^3$	SI
mebi	Mi	$2^{20}$	IEEE 1541
mega	M	$10^6$	SI
gibi	Gi	$2^{30}$	IEEE 1541
giga	G	$10^9$	SI
tebi	Ti	$2^{40}$	IEEE 1541
tera	T	$10^{12}$	SI

La función *modulo* devuelve el resto  $i$  de la división entera entre  $k$  y  $N$ . Esta función se suele denotar de la siguiente forma:

$$i = k \bmod N$$

Otra notación bastante utilizada es:

$$i = k \% N$$

Realizando la división entera de 127 entre 100 se obtiene que el cociente es 1 y el resto es 27.

$$\text{Luego } i = 127 \% 100 = 27.$$

La función *ceil* o función *techo* aproxima el número  $x$  al número entero  $y$  mayor o igual más cercano.

Se denota de la siguiente forma:

$$y = \text{ceil}(x)$$

$$y = \text{ceil}(67,5) = 68.$$

La función *floor* o función *suelo* aproxima el número  $x$  al número entero  $z$  menor o igual más cercano.

Se denota de la siguiente forma:

$$z = \text{floor}(x)$$

$$z = \text{floor}(67,5) = 67.$$

## Preguntas de autoevaluación tema 0

### 0.1. ¿Cuáles son los principales componentes hardware de un computador?

Independientemente del tipo de computador considerado, todos ellos están compuestos por uno o varios de los siguientes tipos de componentes físicos: procesador, memoria principal y controlador o adaptador de E/S.

### 0.2. Señalar cuál es la función del procesador y cuáles son sus principales componentes.

Se encarga de ejecutar secuencialmente las instrucciones de un programa y controlar el funcionamiento de los restantes componentes del computador.

Consta de los siguientes componentes:

- ❖ **Unidad de control.** Se encarga de buscar e interpretar las instrucciones y generar las señales de control necesarias a los restantes elementos del computador para su procesamiento.
- ❖ **Unidad de procesamiento.** Se encarga de la realización de operaciones aritméticas y lógicas sobre los datos.

*Registros.* Permiten al procesador almacenar de forma temporal información necesaria para su funcionamiento.

- ❖ **Unidad de gestión de memoria** (Memory Management Unit, MMU).

### 0.3. Señalar los registros existentes en un procesador.

(Respuesta en sección 0.2.1)

- ❖ **Registro de instrucción.** Contiene la instrucción actual que se tiene que ejecutar.
- ❖ **Registro contador de programa.** Contiene la dirección de memoria donde se encuentra la próxima instrucción que hay que ejecutar.
- ❖ **Puntero de pila.** Dependiendo del tipo de procesador contiene la dirección de memoria de la próxima entrada libre (o de la última entrada ocupada) de la pila.
- ❖ **Registro de estado del procesador.** Contiene diferentes campos para almacenar información del estado del procesador en relación con el programa actualmente en ejecución, como por ejemplo: el modo de ejecución, el nivel de prioridad, el indicador de rebose, el indicador de arrastre, etc.
- ❖ **Registros de propósito general.** Para el almacenamiento de datos y resultados.

#### 0.4. ¿Cuál es la función de la memoria principal de un computador?

Almacenar temporalmente los programas que se van a ejecutar.

#### ¿Con qué tipos de memoria se implementa?

Memorias RAM. Su contenido se puede leer o escribir tantas veces como sea necesario. Su contenido es volátil, es decir, cuando se apaga el computador su contenido se borra.

#### 0.5. ¿Qué es una palabra de memoria?

Es un conjunto ordenado de  $n$  bits contiguos que son manejados como un conjunto por la máquina.

#### 0.6. Conocida la capacidad de la memoria principal expresada en palabras explicar cómo se determina el número de bits de que consta una dirección de memoria.

Numero de palabras x número de bits palabra.

#### 0.7. ¿Qué es un controlador de E/S?

Módulo hardware que permite controlar a uno a varios dispositivos de E/S de un mismo tipo.

#### ¿Puede un controlador de E/S controlar más de un periférico?

Si, si son del mismo tipo.

#### 0.8. ¿Qué es un bus?

Es un medio de comunicación compartido entre dos o más dispositivos.

#### ¿Por qué existen varios tipos de bus en los computadores modernos?

Para solucionar los problemas de rendimiento al ir apareciendo nuevos tipos de dispositivos de E/S con velocidades de transferencias más altas.

#### 0.9. Señalar los diferentes tipos de memoria que conforman la jerarquía de memoria de un computador.

- ❖ **Registros del procesador.** Tiempo de acceso pequeño y capacidad pequeña.
- ❖ **Memoria cache del procesador.** Memoria RAM de acceso asociativo con tiempo de acceso más pequeño que la principal y capacidad también más pequeña.
- ❖ **Memoria principal.** Memoria RAM con mayor capacidad que la cache pero también mayor tiempo de acceso.

- ❖ **Memoria secundaria.** Formada por todos aquellos dispositivos de E/S conectados al computador dedicados al almacenamiento de programas y/o datos.

**0.10. ¿Qué es el repertorio de instrucciones de un procesador?**

Conjunto de instrucciones que es capaz de ejecutar

**¿Puede un procesador ejecutar el repertorio de instrucciones de otro?**

No, solo puede ejecutar instrucciones de su repertorio.

**0.11. Enumerar los diferentes tipos de instrucciones presentes en el repertorio de instrucciones de un procesador.**

- ❖ **Instrucciones de transferencia de datos.** Para copiar o mover datos de una posición de memoria o registro a otra posición de memoria o registro.
- ❖ **Instrucciones de operaciones.** Para realizar sobre operandos almacenados en memoria (o en registros) operaciones aritméticas, lógicas o de desplazamiento de bits.
- ❖ **Instrucciones de transferencia de control.** Para controlar la secuencia de la instrucciones. En esta categoría se encuentran las instrucciones de salto condicional e incondicional.
- ❖ **Instrucciones privilegiadas de gobierno.** Para modificar o comprobar el estado del procesador y los controladores de E/S. También para la gestión de la memoria.

**0.12. Señalar la diferencia entre ejecución en modo supervisor y ejecución en modo usuario.**

En *modo núcleo* el procesador puede ejecutar cualquier instrucción de su repertorio de instrucciones.

En *modo usuario* solo puede ejecutar un conjunto limitado de instrucciones del repertorio. Típicamente las instrucciones privilegiadas no se pueden ejecutar en modo usuario.

**0.13. Explicar qué es el ciclo de ejecución de una instrucción y en qué fases se divide.**

El *ciclo de instrucción* es la secuencia de operaciones realizadas por el procesador durante la ejecución de una instrucción.

Se divide en dos fases:

- ❖ **Fase de búsqueda de la instrucción.** El procesador busca en la memoria, de acuerdo con la dirección almacenada en el registro contador de programa, la instrucción que tiene que ejecutar. A menos que se indique lo contrario el procesador

incrementa el contador de programa después de cada fase de búsqueda para que quede apuntando a la próxima instrucción que hay que ejecutar. La instrucción leída en la memoria se almacena en el registro de instrucción.

- ❖ **Fase de ejecución de la instrucción.** El procesador decodifica el campo código de operación de la instrucción almacenada en el registro de instrucción, interpreta dicho campo de acuerdo con su repertorio de instrucciones y ejecuta las acciones adecuadas.

#### **0.14. Explicar qué es un ensamblador y un compilador.**

Un ensamblador es un programa que toma como entrada un fichero fuente escrito en lenguaje ensamblador y genera como salida un fichero objeto escrito en código máquina.

Un *compilador*, que es un programa que toma como entrada un fichero fuente escrito en un lenguaje de alto nivel y genera como salida un fichero objeto escrito en código máquina.

#### **0.15. Describir las etapas de creación de un fichero ejecutable.**

Con el código fuente mediante un compilador o ensamblador se genera un módulo objeto. Con ese módulo objeto, la librería y otros módulos objetos mediante el editor de enlace se crea el fichero ejecutable.

#### **0.16. ¿Qué es una dirección relativa o lógica?**

Son las direcciones asociadas por el compilador.

#### **¿Y una dirección absoluta o física?**

Son las direcciones de memoria principal.

#### **0.17. Describir el funcionamiento de la E/S controlada por programa e indicar cuál es su principal inconveniente.**

Cuando se ejecuta un programa de gestión de la E/S en el procesador que posee el control completo y directo de la operación de E/S. Este programa, antes de enviar o recibir un dato de un dispositivo de E/S, pregunta al controlador de E/S del dispositivo si éste se encuentra preparado. Dicha pregunta se implementa dentro de un bucle. El programa solo sale del bucle cuando el dispositivo se encuentra listo para enviar o recibir otro dato. A esta forma de implementar una espera por un determinado evento se le denomina *espera activa*.

El principal inconveniente es que desperdicia tiempo de uso del procesador debido a la espera activa del programa de gestión de E/S.

**0.18. Explicar por qué la E/S controlada por interrupciones permite mejorar el rendimiento del computador.**

El controlador de E/S avisará al procesador cuando el dispositivo de E/S estuviese preparado para enviar o recibir un dato. Mientras que el dispositivo de E/S no está listo, el procesador podría ejecutar otros programas, con lo que se mejoraría el rendimiento del computador.

**0.19. Señalar las principales características de una interrupción hardware, una excepción y una trampa.**

- ❖ **Interrupciones hardware.** Son producidas por módulos hardware externos al procesador, como por ejemplo los controladores de E/S. Estos dispositivos puede activar una interrupción para notificar al procesador que se ha completado una operación de E/ S, que se ha producido un cambio en el estado del periférico o que se ha producido algún error en el mismo.
- ❖ **Excepciones.** Son generadas por el hardware al intentar realizar una operación no permitida durante la ejecución de una instrucción de un programa, como por ejemplo, el acceso a una dirección de memoria ilegal, el rebose de una pila, la realización de una división por cero, etc.
- ❖ **Trampas (traps).** Se producen al ejecutar una determinada instrucción privilegiada del repertorio del procesador típicamente para solicitar los servicios del sistema operativo. Las trampas, a diferencia de las excepciones, son insertadas de manera voluntaria por el programador.

**0.20. Describir el funcionamiento del DMA e indicar cuándo se emplea esta técnica de E/S. ¿Puede el DMA afectar al rendimiento del procesador?**

El controlador de E/S puede transferir bloques de datos a la memoria principal (o viceversa) sin la intervención directa del procesador, excepto al inicio y al final de la transferencia. La implementación del DMA requiere de un módulo hardware adicional denominado *controlador de DMA* que se conecta al bus del sistema. Se emplea cuando la cantidad de datos a transferir entre la memoria principal y el dispositivo de E/S es grande.

Si puede afectar al rendimiento del procesador, recuérdese que solo un módulo puede usar el bus del sistema simultáneamente, por lo que si lo usa el controlador de DMA no lo podrá usar el procesador quedándose inactivo a la espera de usarlo.

### 0.21. ¿Qué es un canal o procesador de E/S?

Es un procesador auxiliar que se encarga de realizar todas las operaciones de E/S con un determinado conjunto de dispositivos de E/S.

### 0.22. Describir el proceso de arranque de un computador personal del tipo IBM PC compatible.

Cuando se enciende el computador el procesador comienza a ejecutar el BIOS, que en primer lugar comienza el *autotest de encendido* (Power On Self Test, POST). Si durante el POST se detecta algún problema en el hardware se envían una serie de sonidos por la bocina del computador para avisar al usuario. Una vez realizado el POST, el BIOS busca el dispositivo de E/S desde el que se iniciará el sistema operativo. Para ello consulta en orden secuencial una lista de posibles dispositivos de arranque almacenados en la memoria CMOS de la placa base.

Una vez localizado el dispositivo de arranque, se accede a su primer sector (sector 0) que contiene el registro *de arranque maestro* (Master boot record, MBR), en el que se encuentra almacenado el programa denominado *cargador de arranque* (bootstrap loader) y la *tabla de particiones*, se carga en memoria la principal y se empieza a ejecutar. Accede al sector de arranque de la partición activa (primer sector de dicha partición) para cargar en la memoria principal un pequeño programa denominado *cargador de arranque secundario* que busca en el directorio raíz de dicha partición el código de inicio del sistema operativo lo carga en memoria y le transfiere el control para que se comience a ejecutar. Cuando se inicia el sistema operativo éste obtiene del BIOS la configuración hardware del computador. A continuación comprueba que dispone de los drivers necesarios para cada dispositivo de E/S conectado, y los carga dentro del núcleo. A continuación inicializa las estructuras de datos del sistema operativo. Finalmente lanza un intérprete de comandos o una *interfaz de usuario gráfica* (Graphical User Interface, GUI) para que el usuario pueda interactuar con el sistema operativo.

## Preguntas de autoevaluación tema 1

### 1.1. ¿Qué es un sistema operativo?

Un *sistema operativo* es una capa de software que gestiona de forma eficiente todos los dispositivos hardware de un computador