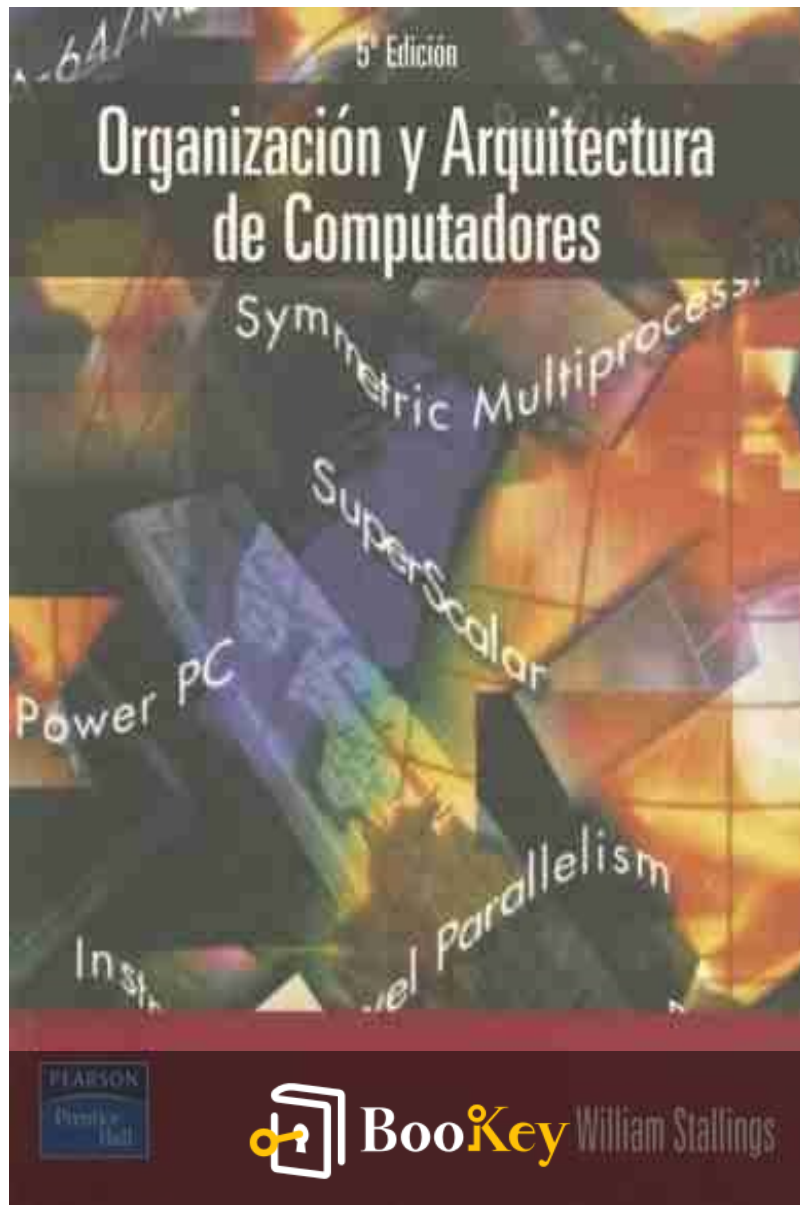


# Organización y arquitectura de computado PDF

William Stallings



Más libros gratuitos en Bookey



Escanear para descargar

# Organización y arquitectura de computado

Fundamentos e Innovaciones en la Arquitectura y  
Diseño Moderno de Computadoras

Escrito por Bookey

[Consulta más sobre el resumen de Organización y  
arquitectura de computado](#)

[Escuchar Organización y arquitectura de computado  
Audiolibro](#)

Más libros gratuitos en Bookey



Escanear para descargar

## Sobre el libro

"Organización y arquitectura de computado" de William Stallings ofrece una exploración profunda y perspicaz de los sistemas de computación contemporáneos, enfatizando principios fundamentales y la importancia del rendimiento en el diseño de computadoras. Sirviendo tanto como una referencia básica como una guía de autoestudio, el libro ilustra eficazmente conceptos a través de numerosos ejemplos prácticos, mostrando tanto arquitecturas modernas CISC como RISC. Organizado en cinco secciones comprensivas—visión general, sistemas de computación, unidades de procesamiento central, unidades de control y organización paralela—es un recurso esencial para ingenieros de computación, arquitectos, profesionales de marketing de productos en empresas tecnológicas y aquellos involucrados en sistemas de información y computación.

Más libros gratuitos en Bookey



Escanear para descargar

# Sobre el autor

William Stallings es un destacado autor estadounidense reconocido por su especialización en ciencias de la computación. Ha escrito numerosos libros de texto que abarcan una variedad de temas, incluyendo sistemas operativos, redes de computadoras, organización y arquitectura de computadoras, y criptografía, realizando aportes significativos al campo.

Más libros gratuitos en Bookey



Escanear para descargar





Escanear para descargar

# Prueba la aplicación Bookey para leer más de 1000 resúmenes de los mejores libros del mundo

Desbloquea de **1000+** títulos, **80+** temas

Nuevos títulos añadidos cada semana



## Perspectivas de los mejores libros del mundo



Prueba gratuita con Bookey



# Lista de contenido del resumen

Capítulo 1 : Organización y arquitectura de computado:

Diseñando para el rendimiento (8ª Edición)

Capítulo 2 : Guía para el

Lector

Capítulo 3 : VISIÓN GENERAL DE LA PRIMERA  
PARTE

Capítulo 4 : PARTE DOS DEL SISTEMA  
INFORMÁTICO

Capítulo 5 : PARTE TRES LA UNIDAD CENTRAL DE PR  
OCESO

Capítulo 6 : PARTE CUATRO LA UNIDAD DE  
CONTROL

Capítulo 7 : PARTE CINCO ORGANIZACIÓN PARALLEL  
A

Más libros gratuitos en Bookey



Escanear para descargar

Capítulo 8 : Apéndice A Proyectos para Enseñar

Organización y Arquitectura de Computado

Capítulo 9 : Apéndice B Lenguaje de Ensamblaje y Temas R  
elacionados

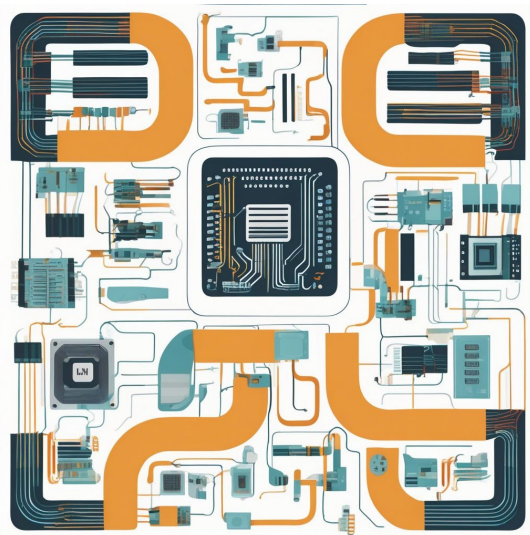
Capítulo 10 : Capítulos en línea

**Más libros gratuitos en Bookey**



Escanear para descargar

# Capítulo 1 Resumen : Organización y arquitectura de computado: Diseñando para el rendimiento (8ª Edición)



## ORGANIZACIÓN Y ARQUITECTURA DE COMPUTADO: DISEÑANDO PARA EL RENDIMIENTO

### Detalles del Autor

- William Stallings
- Octava Edición
- Publicado por Prentice Hall, Upper Saddle River, NJ

Más libros gratuitos en Bookey



Escanear para descargar



## **Información de Derechos de Autor**

- © 2010, 2006 por Pearson Education, Inc.
- Todos los derechos reservados.
- Impreso en los Estados Unidos de América.
- Se requieren permisos para reproducción o transmisión.

## **Equipo Editorial**

- Vicepresidenta y Directora Editorial: Marcia J. Horton
- Editora en Jefe: Michael Hirsch
- Editora Ejecutiva: Tracy Dunkelberger
- Editora Asociada: Melinda Haggerty
- Gerente de Marketing: Erin Davis
- Editora Principal: Scott Disanno
- Editora de Producción: Rose Kernan
- Especialista en Operaciones: Lisa McDowell
- Directora de Arte: Kenny Beck
- Diseño de Portada: Kristine Carney
- Otros gerentes de departamento para derechos, permisos e investigación visual.

## **Información del ISBN**

**Más libros gratuitos en Bookey**



Escanear para descargar

- ISBN-13: 978-0-13-607373-4
- ISBN-10: 0-13-607373-5

## **Dedicatoria**

- A Tricia (ATS), mi amada esposa, la persona más amable y tierna.

**Más libros gratuitos en Bookey**



Escanear para descargar

## Ejemplo

**Punto clave:** Entender la relación entre la arquitectura de computadoras y la optimización del rendimiento es crucial.

**Ejemplo:** Imagina que estás diseñando un nuevo videojuego. Para garantizar que funcione sin problemas en diferentes hardware, necesitas comprender cómo interactúan la CPU, la memoria y el almacenamiento. Al tener en cuenta las capacidades de la arquitectura, puedes optimizar el rendimiento de tu juego: elegir los algoritmos y estructuras de datos correctos mejorará la velocidad y la experiencia del usuario. Esta comprensión de cómo los componentes subyacentes afectan el rendimiento es fundamental para cualquier aspirante a científico de la computación o ingeniero de software.



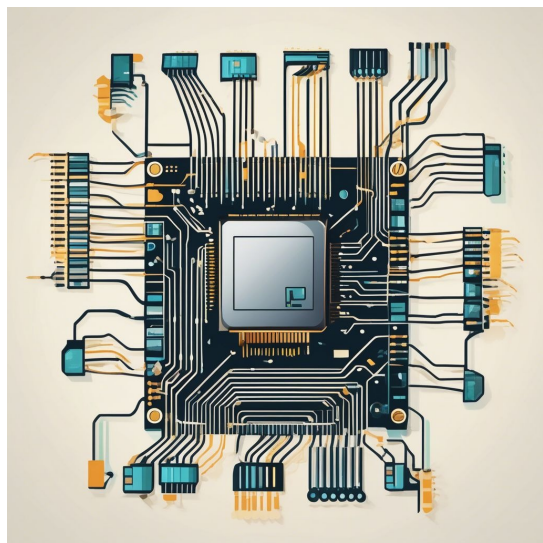
## Pensamiento crítico

**Punto clave:** El enfoque del diseño en el rendimiento de la arquitectura de computadoras es primordial.

**Interpretación crítica:** El énfasis del capítulo en diseñar la arquitectura de computadoras para el rendimiento resalta un aspecto crucial de la tecnología; sin embargo, se debe evaluar críticamente si priorizar el rendimiento puede eclipsar otros factores igualmente vitales, como la eficiencia energética, la escalabilidad o el diseño centrado en el usuario. La búsqueda incesante de rendimiento podría llevar a diseños que no sean sostenibles o adaptables a largo plazo, como argumentan investigadores como Mark A. Horowitz en "El Futuro de la Ingeniería Eléctrica" (IEEE Transactions on Circuits and Systems, 2000). Esta perspectiva invita a una reevaluación de las prioridades de diseño en organización y arquitectura de computadoras.



# Capítulo 2 Resumen : Guía para el Lector



Sección	Contenido
Esquema del Libro	<p>Parte Uno: Visión general de la organización y arquitectura de computado.</p> <p>Parte Dos: Componentes principales, interconexiones, tipos de memoria, entrada/salida y sus relaciones con el sistema operativo.</p> <p>Parte Tres: Arquitectura interna del procesador, aritmética computacional y arquitectura del conjunto de instrucciones.</p> <p>Parte Cuatro: Estructura de la unidad de control y microprogramación.</p> <p>Parte Cinco: Organización paralela, multiprocesadores, clústeres, arquitectura multicore y materiales en línea.</p>
Una Hoja de Ruta para Lectores e Instructores	Enfoque de arriba hacia abajo comenzando con los componentes principales y construyendo la comprensión del diseño del procesador.
¿Por qué estudiar Organización y Arquitectura de Computado?	Materia fundamental para estudiantes de informática e ingeniería; esencial para entender los componentes de computadoras y la selección de sistemas.
Recursos de Internet y Web	<p>Página web dedicada a errores y recursos.</p> <p>Enlaces organizados para matemáticas, guías, investigación, carreras y humor.</p> <p>Sitios web relevantes incluyen la Página Principal de Arquitectura de Computado y Grupos de Interés Especial de ACM.</p>
Grupos de Noticias USENET	Discusiones sobre arquitectura, algoritmos, tecnología de almacenamiento y computación paralela.



## Resumen del Capítulo 2: Guía para el Lector

### Esquema del Libro

El libro se divide en cinco partes:

-

#### **Parte Uno**

: Visión general de la organización y arquitectura de computado, enfocándose en la evolución del diseño de computadoras.

-

#### **Parte Dos**

: Componentes principales de un computador, sus interconexiones, tipos de memoria, I/O, y sus relaciones con los sistemas operativos.

-

#### **Parte Tres**

: Arquitectura interna del procesador, incluyendo la aritmética computacional y la arquitectura del conjunto de instrucciones.

-

#### **Parte Cuatro**

Más libros gratuitos en Bookey



Escanear para descargar



: Enfoque en la estructura de la unidad de control y la microprogramación.

-

## **Parte Cinco**

: Organización paralela, cubriendo multiprocesadores, clústeres, y arquitectura multicore. Se encuentran disponibles capítulos y apéndices adicionales en línea en el sitio web del libro.

El texto enfatiza principios de diseño prácticos y problemas de implementación de arquitecturas de computado contemporáneas utilizando ejemplos principalmente de las arquitecturas Intel x86 y ARM.

## **Una Hoja de Ruta para Lectores e Instructores**

El libro adopta un enfoque de arriba hacia abajo, comenzando con los componentes principales como el procesador, I/O, memoria y dispositivos periféricos. Cada parte se basa en la anterior, mejorando la comprensión del diseño del procesador en su contexto.

## **¿Por qué Estudiar Organización y Arquitectura de Computado?**

**Más libros gratuitos en Bookey**



Escanear para descargar

La arquitectura de computado es un tema fundamental esencial para estudiantes de ciencias de la computación e ingeniería según el Currículo de Computación IEEE/ACM 2001. Comprender los componentes de la computadora, sus características e interacciones es vital para estructurar programas de manera eficiente y tomar decisiones informadas sobre la selección de sistemas.

## **Recursos de Internet y Web**

- Una página web dedicada al libro proporciona erratas y recursos para los lectores.
- Enlaces y documentos adicionales para estudiantes de ciencias de la computación están organizados en categorías como matemáticas, guías prácticas, recursos de investigación, carreras en ciencias de la computación y humor.
- Se mencionan varios sitios relevantes, como la Página de Inicio de Arquitectura de Computado y Grupos de Interés Especial de la ACM.

## **Grupos de Noticias USENET**

Varios grupos de noticias USENET son relevantes para la



organización y arquitectura de computado, ofreciendo discusiones sobre arquitectura, algoritmos aritméticos, tecnología de almacenamiento y computación paralela.

**Más libros gratuitos en Bookey**



Escanear para descargar

## Ejemplo

**Punto clave:** Entender la importancia de la arquitectura en la estructuración de programas conduce a mejores decisiones de diseño.

**Ejemplo:** Imagina que estás desarrollando una nueva aplicación de videojuegos. Al comprender la arquitectura interna de los procesadores, optimizas el rendimiento de tu juego al utilizar eficazmente las capacidades de la CPU, asegurando gráficos más fluidos y tiempos de carga más rápidos, lo que en última instancia mejora la experiencia del usuario.



## Pensamiento crítico

**Punto clave:** El libro enfatiza principios de diseño prácticos y sugiere un enfoque de arriba hacia abajo para el aprendizaje.

**Interpretación crítica:** Si bien la sugerencia del autor de utilizar un enfoque de arriba hacia abajo para entender la arquitectura de computado es loable y se alinea con una pedagogía efectiva, es imperativo cuestionar su universalidad. Esta metodología puede no tener en cuenta los diversos estilos de aprendizaje que se benefician de una técnica más de abajo hacia arriba, particularmente para temas complejos en ciencias de la computación. Muchos estudiantes pueden comprender mejor los detalles intrincados y los conceptos fundamentales cuando pueden explorar los niveles más bajos de la arquitectura antes de integrarlos en sistemas más grandes. La investigación apoya la diversidad de estrategias educativas, indicando que los procesos cognitivos de los estudiantes difieren significativamente, lo que sugiere que un solo enfoque puede no ser adecuado para todos. Por ejemplo, estudios del Journal of Educational Psychology destacan la eficacia de múltiples estrategias de instrucción



adaptadas a las preferencias del aprendiz (Hattie & Donoghue, 2016), desafiando la noción de que un único camino educativo es óptimo. Por lo tanto, aunque la estructura de Stallings ayuda a muchos, se deben explorar perspectivas y metodologías alternativas.



# **Capítulo 3 Resumen : VISIÓN GENERAL DE LA PRIMERA PARTE**

## **PRIMERA PARTE: TEMAS PARA LA PRIMERA PARTE**

El objetivo principal de la Primera Parte es establecer una comprensión básica de la organización y arquitectura de computado, sentando el contexto para el resto del libro.

### **Mapa de Ruta General**

-

#### **Capítulo 1: Introducción**

- Presenta el computador como un sistema jerárquico compuesto por componentes interrelacionados, cada uno con estructuras internas y funciones distintas.

-

#### **Capítulo 2: Evolución y Rendimiento de los Computadores**



- Discute la historia de la tecnología de computadoras y cómo las tendencias en evolución han impactado el diseño de computadores, enfocándose en el rendimiento como un tema central.

## 1.1 Organización y Arquitectura

Este capítulo distingue entre la arquitectura de computado (atributos visibles para el programador que afectan la ejecución) y la organización (unidades operativas y sus interconexiones). Los ejemplos incluyen conjuntos de instrucciones y mecanismos de direccionamiento de memoria como atributos arquitectónicos, contrastados con señales de control y detalles de hardware como atributos organizacionales. Las perspectivas históricas destacan la importancia de cómo una sola arquitectura puede abarcar diferentes diseños organizacionales. ilustrado por la

**Instalar la aplicación Bookey para desbloquear  
texto completo y audio**

Más libros gratuitos en Bookey



Escanear para descargar



Escanear para descargar



# Por qué Bookey es una aplicación imprescindible para los amantes de los libros



## Contenido de 30min

Cuanto más profunda y clara sea la interpretación que proporcionamos, mejor comprensión tendrás de cada título.



## Formato de texto y audio

Absorbe conocimiento incluso en tiempo fragmentado.



## Preguntas

Comprueba si has dominado lo que acabas de aprender.



## Y más

Múltiples voces y fuentes, Mapa mental, Citas, Clips de ideas...

Prueba gratuita con Bookey



# Capítulo 4 Resumen : PARTE DOS DEL SISTEMA INFORMÁTICO

Capítulo	Resumen	Puntos Clave
Parte Dos: Componentes del Sistema Informático	Un sistema informático comprende un procesador, memoria, componentes de I/O y sus interconexiones.	Los detalles del procesador se cubren en la Parte Tres.
Capítulo 3: Una Vista General del Funcionamiento y la Interconexión de la Computadora	Describe los componentes de la computadora y sus interconexiones, enfatizando el diseño del bus del sistema compartido.	<ul style="list-style-type: none"><li>- El ciclo de instrucción incluye la obtención y ejecución.</li><li>- Los componentes se comunican a través de un bus sincronizado.</li><li>- Consideraciones de diseño: arbitraje de bus, temporización, ancho.</li></ul>
Capítulo 4: Memoria Cache	Discute la jerarquía de la memoria y la importancia de la memoria cache para optimizar velocidades.	<ul style="list-style-type: none"><li>- Jerarquía de memoria desde la cache rápida hasta el almacenamiento más lento.</li><li>- La cache almacena datos de acceso frecuente.</li><li>- Algoritmos de reemplazo (por ejemplo, LRU, FIFO) para la gestión de cache.</li></ul>
Capítulo 5: Memoria Interna	Explora los tipos de memoria semiconductor (DRAM, SRAM) y las técnicas de corrección de errores.	<ul style="list-style-type: none"><li>- DRAM requiere refresco; SRAM es más rápida pero más costosa.</li><li>- Códigos de Hamming para detección y corrección de errores.</li></ul>
Capítulo 6: Memoria Externa	Cubre discos magnéticos, tecnología RAID y formatos de memoria óptica.	<ul style="list-style-type: none"><li>- Discos organizados en pistas y sectores; el rendimiento se ve afectado por el tiempo de búsqueda, el retardo rotacional y el tiempo de transferencia.</li><li>- RAID mejora el rendimiento y la fiabilidad.</li><li>- Los formatos ópticos ofrecen opciones de lectura y escritura.</li></ul>
Capítulo 7: Entrada/Salida	Describe el papel del sistema de I/O en la gestión de la comunicación entre dispositivos y el bus del sistema.	<ul style="list-style-type: none"><li>- Enfoques de comunicación: programada, basada en interrupciones y DMA.</li><li>- Las interrupciones manejan I/O asíncrona sin bloqueos de CPU.</li></ul>
Capítulo 8: Soporte del	Discute las funciones del SO y su papel	



Capítulo	Resumen	Puntos Clave
Sistema Operativo	en la gestión de recursos de hardware y la ejecución de software.	<ul style="list-style-type: none"> <li>- Tres niveles de programación optimizan el rendimiento.</li> <li>- Memoria virtual para un uso eficiente de la memoria más allá de los límites físicos.</li> </ul>

## PARTE DOS

### Componentes del Sistema Informático

Un sistema informático está compuesto por un procesador, memoria, componentes de entrada/salida y sus interconexiones. La Parte Dos se centra en cada uno de estos componentes, omitiendo el procesador, que se tratará en la Parte Tres.

### Mapa del Capítulo

-

#### Capítulo 3

: Presenta una visión general del funcionamiento y la interconexión de los ordenadores, discutiendo las interacciones entre componentes.

-



## **Capítulo 4**

: Se centra en el diseño y la organización de la memoria caché.

-

## **Capítulo 5**

: Estudia la memoria interna, incluyendo la organización de la memoria principal, consideraciones de diseño y organizaciones avanzadas de DRAM.

-

## **Capítulo 6**

: Profundiza en el almacenamiento externo, como discos magnéticos, tecnología RAID y memoria óptica.

-

## **Capítulo 7**

: Analiza la organización y operaciones del módulo de entrada/salida (E/S).

-

## **Capítulo 8**

: Discute los sistemas operativos y su apoyo a la funcionalidad del hardware.

## **Capítulo 3: Una Visión General del Funcionamiento y la Interconexión de los Ordenadores**





Describe los componentes esenciales de un ordenador, incluyendo el procesador, la memoria principal y los módulos de E/S, además de las interconexiones que facilitan el intercambio de datos. El texto enfatiza la importancia de un bus de sistema compartido y las consideraciones de diseño para conectar estos componentes de manera eficiente.

## **Puntos Clave**

- El **ciclo de instrucción** incluye fases de obtención y ejecución.
- Los componentes se comunican utilizando un sistema de bus sincronizado.
- Las consideraciones clave de diseño incluyen la arbitraje del bus, el tiempo y el ancho.

---

## **Capítulo 4: Memoria Caché**

### **Visión General**

Los sistemas de memoria de los ordenadores exhiben una



jerarquía caracterizada por diferencias en tipo, tecnología y rendimiento. La memoria caché es crucial para optimizar las velocidades de acceso.

## **Características Clave**

-

### **Jerarquía de Memoria**

: Organiza la memoria en función de la velocidad y la capacidad, desde registros y caché rápidos hasta la memoria principal más lenta y el almacenamiento externo.

-

### **Memoria Caché**

: Almacena datos de acceso frecuente para mejorar el rendimiento del sistema, facilitando un acceso más rápido debido a la localidad de referencia.

## **Elementos del Diseño de Caché**

-

### **Direcciones de Caché**

: Cachés lógicas versus físicas y la estructura de direcciones.

-

### **Funciones de Mapeo**



: Discute los métodos de mapeo directo, asociativo y conjunto asociativo y sus diseños.

## **Algoritmos de Reemplazo**

Existen varias estrategias (por ejemplo, LRU, FIFO) para determinar qué bloque reemplazar cuando la caché está llena.

## **Políticas de Escritura**

-

### **Escritura a Través**

: Todas las actualizaciones se escriben tanto en la caché como en la memoria principal.

-

### **Escritura Retrasada**

: Las actualizaciones solo ocurren en la caché, con cambios enviados a la memoria principal cuando se reemplaza el bloque.

---

## **Capítulo 5: Memoria Interna**



## Tipos de Memoria Semiconductora

-

### **DRAM**

: Utilizada para la memoria principal; requiere actualización debido a fugas de carga.

-

### **SRAM**

: Utilizada para la caché; más rápida y confiable que la DRAM, pero más cara.

## Corrección de Errores

Las técnicas de detección y corrección de errores aseguran la fiabilidad de la memoria, típicamente a través de Códigos de Hamming.

---

## Capítulo 6: Memoria Externa

### Discos Magnéticos

- Esenciales para el almacenamiento de datos; organizados en



pistas y sectores. La velocidad operativa de los discos duros está determinada por el tiempo de búsqueda, el retraso rotacional y el tiempo de transferencia.

## **Tecnología RAID**

- Utiliza múltiples discos para mejorar el rendimiento y la fiabilidad a través de varias configuraciones (por ejemplo, RAID 0 para velocidad, RAID 1 para redundancia).

## **Memoria Óptica**

- Formatos como CD-ROM y DVD proporcionan capacidades de solo lectura, mientras que formatos regrabables (CD-R, CD-RW) permiten el almacenamiento y la eliminación de datos.

---

## **Capítulo 7: Entrada/Salida**

### **Conceptos Básicos del Sistema de E/S**

Los módulos de E/S gestionan la comunicación entre



dispositivos externos y el bus del sistema, empleando enfoques programados, basados en interrupciones y DMA.

## **Interrupciones**

Los mecanismos de interrupción permiten manejar de manera eficiente las solicitudes de E/S asincrónicas sin que la CPU se detenga.

---

## **Capítulo 8: Soporte del Sistema Operativo**

### **Funciones del SO**

El sistema operativo gestiona los recursos de hardware y proporciona una interfaz amigable para la ejecución de programas, la planificación y la gestión de la memoria.

### **Planificación**

Tres niveles de planificación (a largo plazo, a medio plazo y a corto plazo) optimizan el rendimiento del sistema.





## Gestión de Memoria

La memoria virtual permite un uso efectivo de la memoria principal y permite que las aplicaciones superen los límites de la memoria física, confiando en el soporte del hardware.

Más libros gratuitos en Bookey



Escanear para descargar

# Capítulo 5 Resumen : PARTE TRES LA UNIDAD CENTRAL DE PROCESO

## Resumen del Capítulo 5: Organización y arquitectura de computadoras

### La Unidad Central de Proceso

En este capítulo se examinan la estructura y función del procesador, tratado como una "caja negra", destacando componentes clave como los registros, la unidad aritmética y lógica (ALU), las unidades de ejecución de instrucciones y las unidades de control.

### Mapa del Capítulo:

-

### Aritmética de Computadoras (Capítulo 9):

Discute la funcionalidad de la ALU, las representaciones



numéricas (enteras y de punto flotante) e incluye un análisis detallado del estándar de punto flotante IEEE 754.

-

### **Conjuntos de Instrucciones: Características y Funciones (Capítulo 10):**

Cubre los conjuntos de instrucciones de máquina desde la perspectiva de un programador, detallando la semántica del diseño de instrucciones, tipos de operandos y relaciones en lenguaje ensamblador.

-

### **Modos de Direccionamiento (Capítulo 11):**

Se centra en cómo se especifican las direcciones de memoria y los formatos de instrucción utilizados.

-

### **Estructura y Función del Procesador (Capítulo 12):**

Discute la organización de la CPU, registros, ejecución de instrucciones y métodos de segmentación para mejorar el rendimiento.

-

### **Computadoras de Conjuntos de Instrucciones Reducidas (RISC) (Capítulo 13):**

Analiza la arquitectura RISC, enfatizando la simplicidad de los conjuntos de instrucciones, el uso de un gran archivo de registros y la ejecución en paralelo.



## **Temas Clave:**

- Los diferentes diseños y organizaciones de arquitecturas de procesador, incluyendo CISC frente a RISC, examinando las ventajas y desventajas inherentes a varios enfoques.
- Importancia de la segmentación en la ejecución eficiente de instrucciones, junto con mecanismos para lidiar con peligros (peligros de datos, peligros de control) que impactan de manera negativa en el rendimiento.

## **Conclusión:**

El capítulo destaca los aspectos detallados de la arquitectura de computadoras, centrándose en las operaciones internas de las CPU, las operaciones aritméticas y las ganancias de eficiencia alcanzables a través de conjuntos de instrucciones bien diseñados y técnicas de segmentación. A través de la exploración de RISC, el capítulo apunta hacia la evolución del diseño de procesadores, enfatizando la importancia de optimizar la ejecución de instrucciones para cerrar la brecha entre los lenguajes de programación de alto nivel y la arquitectura de máquina.



# Capítulo 6 Resumen : PARTE CUATRO

## LA UNIDAD DE CONTROL

### PARTE CUATRO: TEMAS PARA LA PARTE CUATRO

En la Parte Cuatro, cambiamos nuestro enfoque de las instrucciones de la máquina y las operaciones básicas que ejecuta el procesador a la unidad de control, que orchestra estas operaciones. La unidad de control envía señales de control tanto externamente a la memoria y a los módulos de E/S, como internamente para gestionar el movimiento de datos entre los registros y las funciones de la unidad de control.

### MAPA DEL CAMINO PARA LA PARTE CUATRO

-

### Capítulo 15: Operación de la Unidad de Control

Este capítulo explora cómo las funciones del procesador son



gestionadas por la unidad de control a través de secuencias de microoperaciones, que generan señales de control para la ejecución. Introducimos el concepto de implementación de la unidad de control cableada.

-

## **Capítulo 16: Control Microprogramado**

En este capítulo, examinamos el microprogramado, un método más flexible para la implementación de la unidad de control, que traduce instrucciones en secuencias de microinstrucciones de nivel inferior.

### **PUNTOS CLAVE**

- La ejecución de instrucciones consta de varios ciclos, incluidos los ciclos de búsqueda, ejecución e interrupción, cada uno dividido en microoperaciones más simples.
- La unidad de control asegura que las microoperaciones se

**Instalar la aplicación Bookey para desbloquear  
texto completo y audio**

**Más libros gratuitos en Bookey**



Escanear para descargar



Ad



Escanear para descargar



App Store  
Selección editorial



22k reseñas de 5 estrellas

## Retroalimentación Positiva

Alondra Navarrete

...itas después de cada resumen  
...en a prueba mi comprensión,  
...cen que el proceso de  
...rtido y atractivo."

**¡Fantástico!**



Me sorprende la variedad de libros e idiomas que soporta Bookey. No es solo una aplicación, es una puerta de acceso al conocimiento global. Además, ganar puntos para la caridad es un gran plus!

Beltrán Fuentes

Fi



Lo  
re  
co  
pr

a Vásquez

hábito de  
e y sus  
o que el  
todos.

**¡Me encanta!**



Bookey me ofrece tiempo para repasar las partes importantes de un libro. También me da una idea suficiente de si debo o no comprar la versión completa del libro. ¡Es fácil de usar!

Darian Rosales

**¡Ahorra tiempo!**



Bookey es mi aplicación de  
crecimiento intelectual. Los  
perspicaces y bellamente c  
acceso a un mundo de con

**¡Aplicación increíble!**



encantan los audiolibros pero no siempre tengo tiempo  
escuchar el libro entero. ¡Bookey me permite obtener  
resumen de los puntos destacados del libro que me  
esa! ¡Qué gran concepto! ¡Muy recomendado!

Elvira Jiménez

**Aplicación hermosa**



Esta aplicación es un salvavidas para los a  
los libros con agendas ocupadas. Los resu  
precisos, y los mapas mentales ayudan a  
que he aprendido. ¡Muy recomendable!

Prueba gratuita con Bookey





# Capítulo 7 Resumen : PARTE CINCO

## ORGANIZACIÓN PARALLELA

### PARTE CINCO: TEMAS DE LA PARTE CINCO

Esta sección profundiza en la organización paralela, que implica que varias unidades de procesamiento colaboren para ejecutar aplicaciones de manera efectiva. A diferencia de los procesadores superscalables que manejan la ejecución paralela a nivel de instrucción, el procesamiento paralelo busca niveles más amplios de paralelismo, permitiendo la ejecución concurrente por múltiples procesadores. Surgen desafíos como la coherencia de caché cuando múltiples procesadores con cachés individuales acceden a la memoria compartida, lo que requiere soluciones de hardware o software.

### MAPA DE RUTA PARA LA PARTE CINCO

-



## Capítulo 17: Procesamiento Paralelo

Visión general de las consideraciones del procesamiento paralelo y tres enfoques principales para organizar múltiples procesadores:

-

### **Multiprocesadores Simétricos (SMP)**

: Método común para mejorar el rendimiento y la disponibilidad.

-

### **Clústeres**

: Grupos de computadoras independientes que trabajan en conjunto.

-

### **Acceso a Memoria No Uniforme (NUMA)**

: Estructura comparativamente nueva, con potencial a pesar de su éxito limitado en el mercado.

El capítulo explora además la organización especializada de los procesadores vectoriales.

-

## Capítulo 18: Computadoras Multicore

Describe los chips multicore que encapsulan más de un procesador, mejorando el poder de computación. Se discuten



los problemas de diseño fundamentales asociados con las arquitecturas multicore, usando ejemplos de las arquitecturas x86 y ARM de Intel.

## **PUNTOS CLAVE**

-

### **Estrategias de Ejecución Paralela**

: Las configuraciones comunes incluyen SMPs, clústeres y sistemas NUMA, que mejoran el rendimiento.

-

### **Características de los SMP**

: Involucra procesadores similares con acceso compartido a la memoria principal y dispositivos de E/S, asegurando que cada uno pueda realizar funciones idénticas.

-

### **Multiprocesamiento en Chips**

: Se refiere a la implementación de múltiples procesadores en un solo chip, a menudo agrupados por eficiencia.

-

### **Clústeres**

: Ofrecen escalabilidad y alta disponibilidad, interconectando computadoras completas para actuar como un único recurso.

-



## **Sistemas NUMA**

: Proporcionan memoria compartida con tiempos de acceso variables según la región de acceso, buscando un alto rendimiento entre múltiples procesadores.

## **ORGANIZACIONES DE MÚLTIPLES PROCESADORES**

Varias arquitecturas se clasifican según sus capacidades de procesamiento:

-

### **SISD (Una Instrucción, Un Dato)**

: Sistema uniprocador estándar.

-

### **SIMD (Una Instrucción, Múltiples Datos)**

: Ejecuta una única instrucción sobre múltiples puntos de datos, adecuado para tareas paralelas.

-

### **MISD (Múltiples Instrucciones, Un Dato)**

: Principalmente teórico sin implementación práctica.

-

### **MIMD (Múltiples Instrucciones, Múltiples Datos)**

: Soporta varios procesadores que ejecutan diferentes instrucciones sobre diferentes datos, abarcando SMP,



clústeres y sistemas NUMA.

## **MULTIPROCESADORES SIMÉTRICOS**

Los sistemas SMP mejoran el rendimiento mediante múltiples procesadores similares que comparten memoria y recursos de E/S mientras ejecutan procesos paralelos de manera simultánea, lo cual es crítico para lograr un mayor rendimiento.

## **MULTITHREADING Y MULTIPROCESADORES EN CHIP**

La sección elabora sobre el multithreading como un método para aumentar el rendimiento de instrucciones sin elevar drásticamente la complejidad. Se exploran diferentes diseños de multithreading, destacando cómo amplifican la eficiencia de ejecución.

## **CLÚSTERES**

Se presenta el agrupamiento como un diseño avanzado para manejar el rendimiento y la disponibilidad en tareas



computacionales, permitiendo una escalabilidad orgánica.

## **ACCESO A MEMORIA NO UNIFORME**

Los sistemas NUMA mantienen memoria compartida con tiempos de acceso variables según la región física que se accede, presentando desafíos y consideraciones únicas para la computación de alto rendimiento.

## **COMPUTACIÓN VECTORIAL**

Enfocado en procesadores vectoriales y sistemas de procesamiento de arrays para aplicaciones de alto rendimiento, especialmente en cálculos científicos, con el objetivo de utilizar el paralelismo operacional de manera efectiva.

Para más lecturas y una exploración detallada de los temas discutidos, se citan las referencias y materiales de lectura recomendados dentro de los capítulos.



# **Capítulo 8 Resumen : Apéndice A**

## **Proyectos para Enseñar Organización y Arquitectura de Computado**

### **APÉNDICE A: PROYECTOS PARA ENSEÑAR ORGANIZACIÓN Y ARQUITECTURA DE COMPUTADO**

Muchos instructores creen que los proyectos son esenciales para entender los conceptos de organización y arquitectura de computado. Ayudan a los estudiantes a asimilar conceptos básicos, apreciar el funcionamiento de los sistemas computacionales y construir confianza en su aprendizaje.

#### **Tipos de Proyectos**

El apéndice proporciona orientación sobre seis tipos de proyectos y ejercicios para estudiantes:





- Simulaciones interactivas
- Proyectos de investigación
- Proyectos de simulación
- Proyectos de lenguaje ensamblador
- Tareas de lectura/informe
- Tareas de escritura
- Banco de preguntas

## **A.1 SIMULACIONES INTERACTIVAS**

Se incluyen simulaciones interactivas en esta edición para ayudar a visualizar mecanismos complejos de los sistemas computacionales. Un total de 20 simulaciones ilustran funciones y algoritmos clave, cada una vinculada a secciones relevantes del libro. Las tareas basadas en estas simulaciones se proporcionan a través del Centro de Recursos para Instructores (IRC).

## **A.2 PROYECTOS DE INVESTIGACIÓN**

Los proyectos de investigación implican búsquedas en literatura y en la web, mejorando las habilidades de investigación de los estudiantes y reforzando los conceptos del curso. Los estudiantes pueden trabajar solos o en equipos



y deben presentar una propuesta de proyecto para evaluación. El IRC proporciona formatos y listas de temas para guiar a los estudiantes.

### **A.3 PROYECTOS DE SIMULACIÓN**

Los proyectos de simulación ayudan a los estudiantes a entender las operaciones del procesador y las compensaciones en el diseño. Se recomiendan herramientas como SimpleScalar y SMPCache para simular elementos de procesamiento, permitiendo modificaciones de rendimiento y recopilación de estadísticas detalladas.

- **\*SimpleScalar\***: Un simulador versátil para procesadores modernos disponible sin costo para uso no comercial.
- **\*SMPCache\***: Un simulador basado en trazas enfocado en sistemas de memoria caché con una interfaz intuitiva.

### **A.4 PROYECTOS DE LENGUAJE ENSAMBLADOR**

La programación en lenguaje ensamblador enseña conceptos de hardware de bajo nivel. El lenguaje CodeBlue, diseñado para la enseñanza, facilita el desarrollo de programas en ensamblador en un simulador visual. Los estudiantes



participan en competencias de programación similares a los torneos de Core War, ganando conocimientos sobre los fundamentos de la arquitectura de computadores.

## **A.5 TAREAS DE LECTURA/INFORME**

Las tareas de lectura de la literatura ayudan a reforzar conceptos y brindan experiencia en investigación. El IRC ofrece una lista de artículos sugeridos organizados por capítulo para que los estudiantes analicen.

## **A.6 TAREAS DE ESCRITURA**

Las tareas de escritura mejoran el pensamiento crítico y fomentan una participación más profunda con el tema. El IRC incluye tareas de escritura sugeridas organizadas por capítulo, lo que puede impactar significativamente en los procesos de aprendizaje de los estudiantes.

## **A.7 BANCO DE PREGUNTAS**

Un banco de preguntas está disponible en el sitio del IRC, ofreciendo preguntas de verdadero/falso, de opción múltiple y de rellenar los espacios para cada capítulo, sirviendo como



una herramienta para evaluar la comprensión de los estudiantes.

**Más libros gratuitos en Bookey**



Escanear para descargar

# **Capítulo 9 Resumen : Apéndice B**

## **Lenguaje de Ensamblaje y Temas Relacionados**

### **Lenguaje de Ensamblaje y Temas Relacionados**

#### **1. Visión General del Lenguaje de Ensamblaje**

- El lenguaje de ensamblaje proporciona una representación simbólica del lenguaje máquina específico para un procesador, facilitando la escritura de programas e interacción con el hardware.

#### **2. Ensambladores**

- Se analizan dos tipos de ensambladores: de un paso y de dos pasos. Los ensambladores convierten el lenguaje de ensamblaje en código máquina, siendo el ensamblador de dos pasos más común por su facilidad de comprensión.



### 3. Carga y Vínculo

- El proceso de carga de una aplicación implica colocar el programa en la memoria principal y crear una imagen del proceso. Un enlazador resuelve las referencias entre los módulos compilados.

### 4. Características Clave de los Ensambladores

- El ensamblador traduce el lenguaje de ensamblaje en código máquina e incluye directivas, que son comandos para el ensamblador, en lugar de instrucciones ejecutables por el procesador.

### 5. Elementos del Lenguaje de Ensamblaje

- Las declaraciones típicas en lenguaje de ensamblaje constan de una etiqueta. mnemotécnico. operando(s) y comentarios.

**Instalar la aplicación Bookey para desbloquear  
texto completo y audio**

Más libros gratuitos en Bookey



Escanear para descargar





# Leer, Compartir, Empoderar

Completa tu desafío de lectura, dona libros a los niños africanos.

## El Concepto



×



×



Esta actividad de donación de libros se está llevando a cabo junto con Books For Africa. Lanzamos este proyecto porque compartimos la misma creencia que BFA: Para muchos niños en África, el regalo de libros realmente es un regalo de esperanza.

## La Regla



Gana 100 puntos



Canjea un libro



Dona a África

Tu aprendizaje no solo te brinda conocimiento sino que también te permite ganar puntos para causas benéficas. Por cada 100 puntos que ganes, se donará un libro a África.

Prueba gratuita con Bookey





# Capítulo 10 Resumen : Capítulos en línea

## Capítulo 21: La arquitectura IA-64

### 21.1 Motivación

- IA-64 es una nueva arquitectura de Intel y HP, centrada en el paralelismo a nivel de instrucciones.
- Utiliza técnicas como las Palabras de Instrucción Muy Largas (VLIW), predicación de ramas y carga especulativa.
- EPIC (Computación de Instrucciones Explícitamente Paralelas) encarna esta combinación de ideas.
- La arquitectura representa un cambio significativo respecto a la arquitectura x86 tradicional, diseñada para utilizar de manera más efectiva el creciente número de transistores.

### 21.2 Organización General

- IA-64 emplea un gran número de registros (256 en total: 128 registros de propósito general de 64 bits y 128 registros de punto flotante de 82 bits).
- Anticipa la implementación de ocho o más unidades de



ejecución paralela para procesar instrucciones simultáneamente.

## **21.3 Predicción, Especulación y Pipelining de Software**

- La arquitectura soporta características que promueven el paralelismo a nivel de instrucciones, incluyendo:

-

### **Predicción**

: Elimina ramas ejecutando instrucciones tanto en caminos verdaderos como falsos utilizando registros de predicado.

-

### **Especulación de Control**

: Permite cargas especulativas para manejar la latencia de memoria y difiere las excepciones hasta que sea necesario.

-

### **Especulación de Datos**

: Mueve cargas antes de posibles almacenamientos alteradores si las direcciones no se superponen.

-

### **Pipelining de Software**

: Esta técnica utiliza el desenrollado de bucles para un mejor uso de recursos y procesamiento paralelizado.



## **21.4 Arquitectura del Conjunto de Instrucciones IA-64**

- El formato de instrucción de IA-64 incluye un paquete de 128 bits que contiene hasta tres instrucciones, con un campo de plantilla que indica capacidades de ejecución paralela.
- Las instrucciones tienen un formato de longitud fija de 41 bits que soporta características adicionales como el uso de predicados.

## **21.5 Organización Itanium**

- El procesador Itanium, la primera implementación de IA-64, combina características superscalares con técnicas EPIC.
- Utiliza múltiples unidades de ejecución e incluye canalizaciones sofisticadas para optimizar la ejecución de instrucciones y el manejo de la latencia de memoria.

## **21.6 Lecturas Recomendadas y Sitios Web**

- Las lecturas sugeridas para un estudio más profundo incluyen discusiones sobre la arquitectura EPIC, el pipelining



de software y la mecánica detallada del procesador Itanium.

## 21.7 Términos Clave y Problemas

- Los términos clave incluyen carga avanzada, predicación de ramas, paralelismo explícito, tubería de software y los diversos tipos de registros en la arquitectura IA-64.
- Los problemas que siguen al capítulo desafían al lector a aplicar los conceptos aprendidos a escenarios del mundo real y a la resolución de problemas en la codificación de ensamblador IA-64.

Este resumen describe los conceptos fundamentales y las características innovadoras de la arquitectura IA-64, enfatizando su papel en la mejora del paralelismo a nivel de instrucciones y la optimización del rendimiento en entornos informáticos modernos.







# Las mejores ideas del mundo desbloquean tu potencial

Prueba gratuita con Bookey



Escanear para descargar

# Mejores frases del Organización y arquitectura de computado por William Stallings con números de página

Ver en el sitio web de Bookey y generar imágenes de citas hermosas

## Capítulo 1 | Frases de las páginas 2-4

1. A Tricia (ATS), mi amada esposa, la persona más bondadosa y dulce.

## Capítulo 2 | Frases de las páginas 24-29

1. La computadora es el corazón de la computación.  
Sin ella, la mayoría de las disciplinas de computación hoy en día serían una rama de las matemáticas teóricas.
2. Todos los estudiantes de computación deberían adquirir cierto conocimiento y apreciación de los componentes funcionales de un sistema informático, sus características, su rendimiento y sus interacciones.
3. Es difícil diseñar un sistema operativo bien sin conocimiento de la arquitectura subyacente.
4. Una buena comprensión general de estos conceptos será



útil tanto en otras áreas de estudio como en el trabajo futuro después de la graduación.

### **Capítulo 3 | Frases de las páginas 30-85**

1. La intención de este libro es proporcionar una discusión exhaustiva sobre los fundamentos de la organización y arquitectura de computado y relacionarlos con los problemas de diseño computacional contemporáneos.
2. Sin duda, la aplicación de estos conceptos depende del estado actual de la tecnología y los objetivos de coste/rendimiento del diseñador.
3. La naturaleza jerárquica de los sistemas complejos es esencial tanto para su diseño como para su descripción.
4. En última instancia, este control es ejercido por el/los individuo(s) que proporciona(n) al computador instrucciones.
5. El diseñador solo necesita ocuparse de un nivel particular del sistema a la vez.





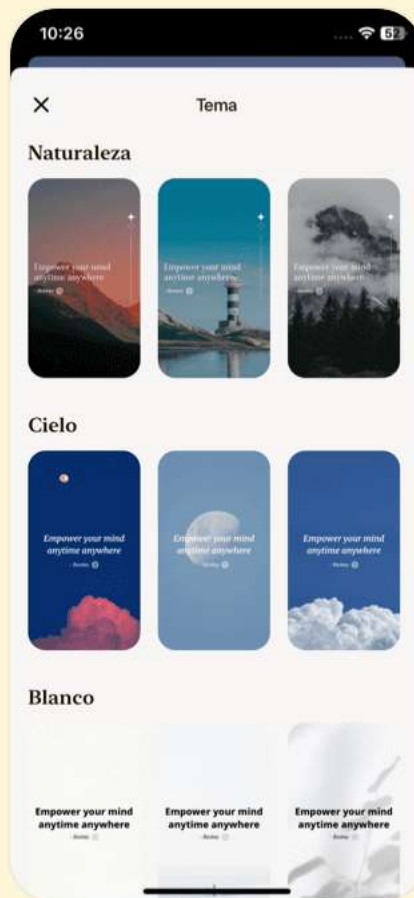


Descarga la app Bookey para disfrutar

# Más de 1 millón de citas Resúmenes de más de 1000 libros

**¡Prueba gratuita disponible!**

Escanear para descargar



## Capítulo 4 | Frases de las páginas 86-325

- 1.El sistema operativo permite que los recursos del sistema informático se utilicen de manera eficiente.
- 2.El sistema operativo oculta los detalles del hardware al programador y le proporciona una interfaz conveniente para utilizar el sistema.
- 3.El sistema operativo es responsable de gestionar estos recursos.
- 4.En un entorno transaccional, el rendimiento se ve afectado.
- 5.Con múltiples discos, las solicitudes de E/S separadas pueden manejarse en paralelo, siempre que los datos requeridos residan en discos separados.
- 6.RAID 0, como todos los niveles de RAID, va más allá de simplemente distribuir los datos a través de un arreglo de discos: Los datos se distribuyen entre los discos disponibles.
- 7.El retraso rotacional de un disco será de 1.5 ms.
- 8.Una ventaja clave del diseño de caché dividida es que elimina la contención por la caché entre la unidad de



obtención/decodificación de instrucciones y la unidad de ejecución.

## **Capítulo 5 | Frases de las páginas 326-581**

1. Una referencia del operando en una instrucción contiene ya sea el valor real del operando (inmediato) o una referencia a la dirección del operando.
2. Un formato de instrucción simple también permite una decodificación de instrucciones más fácil, ya que se involucran menos bits y los datos se pueden acceder rápidamente sin cálculos complicados.
3. El uso de direccionamiento inmediato elimina la necesidad de una referencia de memoria, ahorrando tiempo durante la ejecución de instrucciones.
4. El diseño del conjunto de instrucciones PDP-11 se centró en proporcionar una amplia variedad de modos de direccionamiento, permitiendo una mayor flexibilidad en cómo las instrucciones pueden manipular datos.
5. Las arquitecturas RISC priorizan un conjunto de



instrucciones limitado que está optimizado para un modelo de ejecución por tuberías, simplificando la lógica de control y mejorando el rendimiento.

6. Los predictores de ramas dinámicos utilizan el comportamiento histórico para realizar predicciones más precisas respecto a la dirección de las ramas condicionales en la ejecución del programa.
7. Con las arquitecturas de carga/almacenamiento, todas las operaciones ocurren entre registros, asegurando un procesamiento de datos más rápido ya que no hay accesos directos a la memoria durante las tareas computacionales.
8. La ejecución condicional de instrucciones reduce la necesidad de múltiples instrucciones de rama, simplificando así el flujo del pipeline de instrucciones y mejorando el rendimiento general.
9. La unidad de control gestiona el flujo de datos e instrucciones dentro y fuera del procesador, coordinando la operación de la ALU y varios registros.
10. Las arquitecturas RISC a menudo logran un mejor



equilibrio entre la simplicidad de las instrucciones y la velocidad de ejecución que sus contrapartes CISC al aprovechar un mayor número de registros y formatos de instrucción más simples.

## **Capítulo 6 | Frases de las páginas 582-649**

1. La unidad de control es la parte del procesador que realmente hace que las cosas sucedan.
2. La ejecución de un programa consiste en operaciones que involucran estos elementos del procesador.
3. La unidad de control hace que el procesador pase por una serie de microoperaciones en la secuencia apropiada.
4. La ejecución de una instrucción implica la ejecución de una secuencia de subpasos, generalmente llamados ciclos.
5. Un último punto que vale la pena mencionar es que una de las microoperaciones implica una suma.





Descarga la app Bookey para disfrutar

# Más de 1 millón de citas Resúmenes de más de 1000 libros

**¡Prueba gratuita disponible!**

Escanear para descargar





## Capítulo 7 | Frases de las páginas 650-729

1. A medida que la tecnología informática ha evolucionado y que el costo del hardware de computadoras ha disminuido, los diseñadores de computadoras han buscado cada vez más oportunidades para el paralelismo, generalmente para mejorar el rendimiento y, en algunos casos, para aumentar la disponibilidad.
2. Un clúster es un grupo de computadoras interconectadas, trabajando juntas como un recurso informático unificado que puede crear la ilusión de ser una sola máquina.
3. En una organización SMP, múltiples procesadores comparten una memoria común. Esta organización plantea el problema de la coherencia de caché, al cual se dedica una sección separada.
4. Con un caché L2 compartido, el problema de la coherencia de caché se limita al nivel de caché L1, lo que puede proporcionar alguna ventaja de rendimiento adicional.
5. La SCU monitorea las operaciones sobre los datos





compartidos para optimizar la migración del estado MESI.

## **Capítulo 8 | Frases de las páginas 730-735**

- 1.Sin proyectos, puede ser difícil para los estudiantes entender algunos de los conceptos básicos y las interacciones entre los componentes.
- 2.Los proyectos refuerzan los conceptos introducidos en el libro, brindan a los estudiantes una mayor apreciación del funcionamiento interno de los procesadores y los sistemas computacionales, y pueden motivar a los estudiantes y darles confianza en que han dominado el material.
- 3.Las simulaciones interactivas proporcionan una herramienta poderosa para comprender las características de diseño complejas de un sistema informático moderno.
- 4.Lo mejor es requerir algún tipo de propuesta de proyecto al principio del curso, dando al instructor tiempo para evaluar la propuesta en cuanto al tema adecuado y al nivel de esfuerzo apropiado.
- 5.La programación en lenguaje ensamblador se utiliza a menudo para enseñar a los estudiantes componentes de



hardware de bajo nivel y conceptos básicos de arquitectura de computadoras.

6.Las tareas de escritura pueden tener un poderoso efecto multiplicador en el proceso de aprendizaje en una disciplina técnica como las comunicaciones de datos y las redes.

## **Capítulo 9 | Frases de las páginas 736-762**

- 1.El lenguaje ensamblador es un lenguaje de programación que está a un paso del lenguaje de máquina.
- 2.Un ensamblador es un programa que traduce lenguaje ensamblador a código máquina.
- 3.El primer paso en la creación de un proceso activo es cargar un programa en la memoria principal y crear una imagen del proceso.
- 4.Comprender la programación en lenguaje ensamblador hace que los estudiantes sean mejores programadores en lenguajes de alto nivel (HLL).
- 5.El ensamblador no está verificando si se cumplen las



convenciones de llamada y las convenciones de guardado de registros.

**Más libros gratuitos en Bookey**



Escanear para descargar



Descarga la app Bookey para disfrutar

# Más de 1 millón de citas Resúmenes de más de 1000 libros

**¡Prueba gratuita disponible!**

Escanear para descargar



## Capítulo 10 | Frases de las páginas -846

1. Debido a la naturaleza binaria inherente de los componentes de los ordenadores digitales, todas las formas de datos dentro de los ordenadores se representan mediante varios códigos binarios.
2. El procesador utiliza esta información para realizar ejecuciones en paralelo.
3. Para cualquier función dada, hay una serie de realizaciones alternativas.
4. El corazón de este nuevo enfoque es el concepto de paralelismo explícito.
5. La predicación es una técnica mediante la cual el compilador determina qué instrucciones pueden ejecutarse en paralelo.





Descarga la app Bookey para disfrutar

# Más de 1 millón de citas Resúmenes de más de 1000 libros

**¡Prueba gratuita disponible!**

Escanear para descargar



# Organización y arquitectura de computado Preguntas

Ver en el sitio web de Bookey

## Capítulo 1 | Organización y arquitectura de computado: Diseñando para el rendimiento (8ª Edición)| Preguntas y respuestas

### 1.Pregunta

**¿Cuáles son los elementos clave de la organización y arquitectura de computado según William Stallings?**

Respuesta: Los elementos clave incluyen la

estructura y el comportamiento de los sistemas de computado, los principios de diseño utilizados para el rendimiento y cómo los componentes de hardware trabajan juntos para ejecutar el software de manera eficiente.

### 2.Pregunta

**¿Por qué es importante el rendimiento en la organización y arquitectura de computado?**

Respuesta: El rendimiento es crucial porque determina cuán rápida y eficazmente una computadora puede completar

Más libros gratuitos en Bookey



Escanear para descargar



tareas, impactando la experiencia del usuario y la eficiencia general del sistema.

### 3.Pregunta

**¿Cómo afecta la disposición de un sistema de computado a su organización y arquitectura?**

Respuesta:La disposición, o organización, de un sistema de computado afecta la velocidad de procesamiento de datos, la facilidad de acceso a la memoria y la forma en que interactúan los diferentes componentes, influyendo directamente en el rendimiento del sistema.

### 4.Pregunta

**¿Puedes dar un ejemplo de un compromiso en el diseño de arquitectura de computado?**

Respuesta:Un notable compromiso en el diseño de arquitectura de computado es entre el costo y el rendimiento; un rendimiento más alto a menudo puede requerir componentes más costosos, lo que lleva a restricciones presupuestarias para los desarrolladores.

### 5.Pregunta

**¿Cuál es la importancia de entender la arquitectura de**



## **computado en el desarrollo de software?**

Respuesta:Entender la arquitectura de computado es importante para el desarrollo de software porque informa a los desarrolladores sobre cómo el software interactúa con el hardware, lo que conduce a un código más eficiente que puede aprovechar las capacidades del sistema.

### **6.Pregunta**

**¿De qué maneras los avances en tecnología desafían los principios existentes de organización de computado?**

Respuesta:Los avances en tecnología, como los procesadores multicore y la computación en la nube, desafían los principios existentes al exigir una utilización más eficiente de los recursos y necessitar nuevos diseños de arquitectura para manejar cargas de datos incrementadas.

### **7.Pregunta**

**¿Cómo podrían los contenidos de este capítulo estar conectados con temas futuros en la ingeniería de computado?**

Respuesta:Este capítulo sienta las bases para conceptos más



profundos en la ingeniería de computado, como soluciones de almacenamiento de datos, arquitectura de redes y el impacto de tecnologías emergentes como la computación cuántica en los principios arquitectónicos actuales.

## **Capítulo 2 | Guía para el Lector**

### **Preguntas y respuestas**

#### **1.Pregunta**

**¿Por qué es importante estudiar organización y arquitectura de computado?**

Respuesta: Estudiar organización y arquitectura de computado es crucial porque permite a las personas comprender los componentes funcionales de los sistemas de computación, sus características de rendimiento y sus interacciones. Este conocimiento es esencial para tomar decisiones informadas sobre el diseño del sistema, mejorar la eficiencia en la programación y entender la relación entre el software y el hardware.

#### **2.Pregunta**

**Más libros gratuitos en Bookey**



Escanear para descargar

## **¿Cuáles son los componentes clave de la arquitectura de computado que los estudiantes deberían aprender?**

Respuesta: Los estudiantes deberían aprender sobre componentes clave como el procesador, la memoria, los sistemas de entrada/salida y los dispositivos periféricos. Cada uno de estos componentes desempeña un papel crítico en la funcionalidad general de los sistemas de computado.

### **3.Pregunta**

## **¿Qué distingue la arquitectura Intel x86 de la arquitectura ARM?**

Respuesta: La arquitectura Intel x86 es principalmente un Computador de Conjunto de Instrucciones Complejas (CISC) que incorpora ciertas características de los Computadores de Conjunto de Instrucciones Reducidas (RISC), mientras que la arquitectura ARM es fundamentalmente un RISC. La arquitectura x86 es conocida por sus capacidades de instrucciones complejas, mientras que ARM se centra en conjuntos de instrucciones más simples y eficientes.

### **4.Pregunta**



## **¿Qué enfoque tiene este libro en su presentación de la arquitectura de computado?**

Respuesta:El libro sigue un enfoque de arriba hacia abajo, comenzando con conceptos de nivel superior como los principales componentes del sistema de computado y avanzando hacia temas más detallados relacionados con la arquitectura interna y el procesamiento. Esta estructura ayuda a los lectores a entender los requisitos externos que impulsan las decisiones de diseño.

## **5.Pregunta**

## **¿Cómo beneficia a los estudiantes entender la arquitectura de computado en sus futuras carreras?**

Respuesta:Una comprensión integral de la arquitectura de computado equipa a los estudiantes con las habilidades para elegir sistemas apropiados para tareas específicas, optimizar el rendimiento de programas y diseñar sistemas de software efectivos. Este conocimiento práctico es aplicable en diversos campos de la computación, asegurando que los graduados puedan adaptarse a una amplia gama de roles en la



industria.

## 6.Pregunta

**¿Qué recursos están disponibles para estudiantes y profesionales que estudian arquitectura de computado?**

Respuesta: Numerosos recursos en línea, incluyendo páginas web dedicadas al libro, listas de erratas por precisión y artículos académicos adicionales y sitios relacionados con la arquitectura de computado, proporcionan un amplio apoyo para el aprendizaje continuo.

## 7.Pregunta

**¿Puedes dar un ejemplo de una aplicación práctica del conocimiento en arquitectura de computado?**

Respuesta: Un ejemplo práctico es cuando un ingeniero de computado tiene la tarea de seleccionar un sistema de computado rentable para una gran organización. Comprender los compromisos entre diferentes componentes como la velocidad de la CPU y el tamaño de la memoria es vital para tomar una decisión informada que equilibre el rendimiento con el presupuesto.



## 8.Pregunta

**¿Por qué es vital que los ingenieros de software tengan conocimiento de la arquitectura de computado?**

Respuesta: Los ingenieros de software se benefician de entender la arquitectura de computado porque les permite escribir programas que hacen un uso óptimo del hardware, lo que lleva a un mejor rendimiento. El conocimiento de la arquitectura subyacente les ayuda a estructurar su código para aprovechar las capacidades de la CPU y las características de gestión de memoria.

## 9.Pregunta

**¿Qué papel juegan los dispositivos periféricos en la arquitectura de computado?**

Respuesta: Los dispositivos periféricos son esenciales en la arquitectura de computado ya que permiten la interacción externa con el sistema de computado, afectando la eficiencia y funcionalidad general. Entender cómo estos dispositivos se conectan y comunican con el procesador es clave para diseñar experiencias de usuario sin problemas.





## 10.Pregunta

**¿De qué manera puede el estudio de organización y arquitectura de computado reforzar conceptos en otras áreas de la computación?**

Respuesta: Estudiar organización y arquitectura de computado ayuda a reforzar principios comunes en otras disciplinas de computación, como lenguajes de programación y sistemas operativos, al ilustrar cómo las decisiones arquitectónicas impactan el desarrollo de software y la funcionalidad del sistema.

## **Capítulo 3 | VISIÓN GENERAL DE LA PRIMERA PARTE** **| Preguntas y respuestas**

### 1.Pregunta

**¿Cuáles son las distinciones fundamentales entre la arquitectura de computadoras y la organización de computadoras?**

Respuesta: La arquitectura de computadoras se refiere a los atributos de un sistema informático que son visibles para el programador, afectando la



ejecución lógica de los programas, como el conjunto de instrucciones y los métodos de direccionamiento de memoria. En cambio, la organización de computadoras incluye las unidades operativas y sus interconexiones que realizan estas especificaciones arquitectónicas, que pueden ser transparentes para el programador, como las señales de control y la tecnología de memoria física.

## 2.Pregunta

**¿Por qué es importante entender la naturaleza jerárquica de los sistemas informáticos en la arquitectura de computadoras?**

Respuesta:La organización jerárquica permite a los diseñadores abordar un sistema informático en niveles, simplificando tanto el diseño como el análisis. Al descomponer sistemas complejos en subsistemas manejables en varios niveles, los diseñadores pueden centrarse en aspectos particulares sin perder de vista la funcionalidad general del sistema.



### 3.Pregunta

**¿Puedes explicar las funciones que puede realizar una computadora?**

Respuesta:En general, una computadora puede realizar cuatro funciones principales: procesamiento de datos, almacenamiento, movimiento y control. El procesamiento de datos implica manipular datos para convertirlos en información significativa; el almacenamiento se refiere a guardar datos para uso inmediato o futuro; el movimiento de datos implica transferir datos entre la computadora y sus periféricos u otros sistemas; el control gestiona cómo se realizan estas operaciones.

### 4.Pregunta

**¿Qué importancia histórica tiene la arquitectura IBM System/370 en relación con la compatibilidad y los avances en los sistemas informáticos?**

Respuesta:Introducida en 1970, la arquitectura IBM System/370 ejemplificó la longevidad y flexibilidad en la computación. Permitió a los clientes actualizar a modelos más nuevos sin desechar el software existente, protegiendo la



inversión de los usuarios. Esta práctica sentó un precedente para la compatibilidad hacia atrás en futuros diseños, apoyando una transición más suave a medida que la tecnología evolucionaba.

## 5.Pregunta

**¿Cómo ha transformado la introducción de la microelectrónica el diseño y la funcionalidad de las computadoras?**

Respuesta:La microelectrónica revolucionó el diseño de computadoras al permitir la integración de más componentes en un solo chip, reduciendo drásticamente el tamaño, costo y consumo de energía, mientras se mejora el rendimiento. Este cambio llevó a la proliferación de computadoras personales y sistemas integrados, haciendo que la computación potente sea ampliamente accesible.

## 6.Pregunta

**¿Cuáles son las implicaciones de la Ley de Moore en el contexto de la computación moderna?**

Respuesta:La Ley de Moore predice la duplicación de transistores en circuitos integrados aproximadamente cada



dos años, lo que lleva a un aumento del rendimiento y una disminución de costos. Esto ha impulsado avances en la velocidad del procesador, la capacidad de memoria y las funcionalidades generales de las computadoras, mejorando continuamente las capacidades de los sistemas informáticos modernos.

## 7.Pregunta

**¿Cómo proporcionan las evaluaciones de rendimiento como MIPS y puntos de referencia información sobre las capacidades de diferentes procesadores?**

Respuesta:Las medidas de rendimiento como MIPS

(Millones de Instrucciones por Segundo) y puntos de referencia estandarizados evalúan cuán eficientemente las CPU ejecutan instrucciones bajo varias cargas de trabajo. Al comparar el rendimiento a través de diferentes arquitecturas utilizando cargas de trabajo consistentes, estas evaluaciones ayudan a identificar qué procesadores sobresalen en tareas específicas, permitiendo a consumidores y desarrolladores tomar decisiones informadas.



## 8.Pregunta

**¿Qué papel juega la Ley de Amdahl en la comprensión de las limitaciones de la mejora del rendimiento en los sistemas informáticos?**

Respuesta:La Ley de Amdahl ilustra que las mejoras en un aspecto de un sistema, como el aumento de la velocidad de procesamiento a través de la paralelización, no llevan a mejoras proporcionales en el rendimiento general si hay partes del proceso que permanecen en serie. Este principio destaca los rendimientos decrecientes encontrados al mejorar componentes paralelos sin abordar limitaciones en serie.

## 9.Pregunta

**¿Cuál podría ser el impacto de múltiples procesadores en un solo chip en términos de rendimiento?**

Respuesta:Implementar múltiples procesadores (multicore) en un solo chip puede mejorar significativamente el rendimiento sin aumentar la velocidad del reloj. Esta elección arquitectónica permite el procesamiento paralelo, duplicando o triplicando efectivamente el rendimiento siempre que el software esté optimizado para utilizar los núcleos



adicionales, lo que lleva en última instancia a una ejecución más eficiente de las tareas.

### 10.Pregunta

**¿Cómo refleja la evolución de los componentes y la arquitectura de las computadoras cambios más amplios en la tecnología y la sociedad?**

Respuesta:La evolución de los componentes de las computadoras, desde tubos de vacío hasta transistores y circuitos integrados, paralela los avances en la ciencia de materiales, procesos de fabricación y una creciente dependencia social de la tecnología. Cada transición no solo ha hecho que las computadoras sean más accesibles y potentes, sino que también ha impulsado desarrollos en campos como las telecomunicaciones, la atención médica y la educación, redefiniendo la vida moderna.

### 11.Pregunta

**¿Por qué es crítica la distinción entre las unidades de control y las ALUs para entender la funcionalidad de una CPU?**

Respuesta:La unidad de control orquesta las operaciones de





una CPU, gestionando el flujo de datos e instrucciones, mientras que la ALU (Unidad Aritmética Lógica) realiza los cálculos reales y las operaciones lógicas. Comprender esta distinción es vital, ya que ayuda a aclarar cómo las CPU ejecutan instrucciones, cómo se procesan los datos y cómo interactúan los diferentes componentes dentro de un sistema informático.

**Más libros gratuitos en Bookey**



Escanear para descargar



Escanear para descargar

# Prueba la aplicación Bookey para leer más de 1000 resúmenes de los mejores libros del mundo

Desbloquea de **1000+** títulos, **80+** temas

Nuevos títulos añadidos cada semana

- Brand
- Liderazgo & Colaboración
- Gestión del tiempo
- Relaciones & Comunicación
- Kn
- ategia Empresarial
- Creatividad
- Memorias
- Dinero e Inversiones
- Conózcase a sí mismo
- prendimiento
- Historia del mundo
- Comunicación entre Padres e Hijos
- Autocuidado
- M

## Perspectivas de los mejores libros del mundo



Prueba gratuita con Bookey



# Capítulo 4 | PARTE DOS DEL SISTEMA INFORMÁTICO

## Preguntas y respuestas

### 1.Pregunta

**¿Cuáles son los elementos clave involucrados en la organización y operación de un sistema de memoria de computadora, según se discute en el capítulo 4 del libro?**

Respuesta: Los elementos clave involucrados en la organización y operación de un sistema de memoria de computadora incluyen:

1. Jerarquía de memoria - Organización de los componentes de memoria en niveles basados en velocidad, capacidad y costo.
2. Memoria caché - Diseño y estructura de la memoria caché, incluidos el uso de cachés de código y de datos por separado.
3. Tipos de memoria - Comprensión de los diferentes tipos de memoria como SRAM, DRAM, ROM y sus características respectivas.
4. Técnicas de corrección de errores - Mecanismos



para detectar y corregir errores en el almacenamiento de memoria, incluidos los códigos de Hamming y ECC.

5. Tecnologías avanzadas de memoria - DRAM sincrónica, DDR SDRAM y sus mejoras sobre la DRAM tradicional.

## 2.Pregunta

**¿Cómo mejora la memoria caché el rendimiento de un sistema informático?**

Respuesta:La memoria caché mejora el rendimiento al almacenar copias de datos a los que se accede con frecuencia desde la memoria principal, basándose en el principio de localidad de referencia, que establece que los programas tienden a acceder repetidamente a una cantidad limitada de datos en un corto período de tiempo. Al tener estos datos disponibles en caché, la CPU puede acceder a ellos mucho más rápido que si tuviera que recuperarlos de la memoria principal más lenta.

## 3.Pregunta

Más libros gratuitos en Bookey



Escanear para descargar



## **¿Cuáles son las principales diferencias entre la RAM dinámica (DRAM) y la RAM estática (SRAM)?**

Respuesta: Las principales diferencias entre DRAM y SRAM son:

1. Estructura: DRAM utiliza capacitores para almacenar bits, lo que requiere refresco, mientras que SRAM utiliza flip-flops para almacenar bits, manteniendo los datos sin necesidad de refresco.
  2. Velocidad: SRAM es generalmente más rápida que DRAM debido a su estructura de acceso más simple y rápida.
  3. Densidad: DRAM puede ser empaquetada con mayor densidad que SRAM, lo que la hace más barata por bit.
- SRAM se utiliza para memoria caché, mientras que DRAM se utiliza típicamente para memoria principal.

## **4.Pregunta**

### **¿Qué tipos de sistemas operativos existen, según la interacción del usuario y la gestión de recursos?**

Respuesta: Los sistemas operativos se pueden clasificar según:



## 1. Interacción del usuario:

- Sistemas operativos por lotes: Procesan trabajos secuencialmente sin interacción del usuario.
- Sistemas operativos interactivos: Permiten la interacción directa del usuario durante la ejecución de trabajos.

## 2. Gestión de recursos:

- Sistemas de multiprogramación: Permiten que múltiples programas estén en memoria simultáneamente, compartiendo la CPU.
- Sistemas de uniprogramación: Ejecutan un programa a la vez, utilizando los recursos de manera secuencial.

## 5.Pregunta

**¿Cuál es la importancia de la gestión de memoria y la programación de procesos en los sistemas operativos?**

Respuesta:La gestión de memoria es crucial porque determina cómo se asigna la memoria a los procesos y asegura una utilización eficiente de la memoria del sistema, previniendo fugas de memoria y sobrescrituras. La programación de procesos es esencial para gestionar la



ejecución de múltiples procesos, asignando el tiempo de CPU de manera efectiva y asegurando que todos los procesos reciban un acceso justo a los recursos de la CPU, lo cual es particularmente importante en un entorno de multiprogramación.

## 6.Pregunta

**¿Puedes explicar el concepto de RAID y sus varios niveles?**

Respuesta:RAID (Conjunto Redundante de Discos Independientes) es una tecnología de virtualización de almacenamiento de datos que combina múltiples componentes de unidades de disco físico en una única unidad lógica para redundancia de datos y mejoras en el rendimiento. Los niveles incluyen:

1. RAID 0: Sin redundancia, distribuye datos entre múltiples discos (mejora el rendimiento).
2. RAID 1: Datos espejados en dos discos (redundancia).
3. RAID 2: Distribución a nivel de bit con código de Hamming.





4. RAID 3: Distribución a nivel de byte con disco de paridad dedicado.
5. RAID 4: Distribución a nivel de bloque con disco de paridad dedicado.
6. RAID 5: Distribución a nivel de bloque con paridad distribuida.
7. RAID 6: Doble paridad distribuida para mayor tolerancia a fallos.

## **Capítulo 5 | PARTE TRES LA UNIDAD CENTRAL DE PROCESO**

### **| Preguntas y respuestas**

#### **1.Pregunta**

**¿Qué tipos de operaciones se incluyen típicamente en una arquitectura RISC?**

Respuesta: Las arquitecturas RISC suelen incluir operaciones de transferencia de datos, operaciones aritméticas, operaciones lógicas, instrucciones de movimiento de datos y operaciones de control como las instrucciones de salto.

#### **2.Pregunta**

**Más libros gratuitos en Bookey**



Escanear para descargar

## **¿En qué se diferencia el direccionamiento inmediato del direccionamiento indirecto?**

Respuesta:El direccionamiento inmediato incluye el operando directamente dentro de la instrucción, mientras que el direccionamiento indirecto hace referencia a la dirección del operando ubicado en la memoria.

### **3.Pregunta**

## **¿Por qué son importantes los códigos de condición en el diseño de un procesador?**

Respuesta:Los códigos de condición indican el estado del procesador tras una operación, lo que puede ser utilizado para tomar decisiones en las instrucciones de salto.

### **4.Pregunta**

## **¿Cuál es la importancia de utilizar un archivo de registros grande en la arquitectura RISC?**

Respuesta:Utilizar un archivo de registros grande permite un acceso más rápido a los operandos, minimiza las referencias de memoria y mejora el rendimiento gracias a la localidad de la referencia.



## 5.Pregunta

**¿Cómo mejora el rendimiento de un procesador el pipelining?**

Respuesta:El pipelining permite que múltiples fases de instrucción se procesen simultáneamente, aumentando así el rendimiento de instrucciones.

## 6.Pregunta

**¿Cuál es el papel del contador de programa (PC)?**

Respuesta:El contador de programa contiene la dirección de la siguiente instrucción a ejecutar y se actualiza después de cada recuperación de instrucción.

## 7.Pregunta

**¿Qué desafíos enfrenta el pipelining de instrucciones con las instrucciones de salto?**

Respuesta:El pipelining puede enfrentar problemas debido a la incertidumbre sobre si se toma un salto, lo que puede llevar a que se recuperen instrucciones incorrectas, requiriendo pausas o la limpieza del pipeline.

## 8.Pregunta

**Describe la diferencia entre la forma en que las**



**arquitecturas RISC y CISC manejan las instrucciones.**

Respuesta: Las arquitecturas RISC utilizan un conjunto más pequeño y simple de instrucciones que a menudo se ejecutan en un ciclo de reloj, mientras que las arquitecturas CISC tienden a tener un conjunto más amplio de instrucciones complejas que pueden requerir múltiples ciclos para ejecutarse.

### **9.Pregunta**

**En el contexto de la arquitectura ARM, ¿cuál es la importancia de que el conjunto de instrucciones de ARM sea mayormente condicional?**

Respuesta: La naturaleza condicional de las instrucciones ARM permite una ejecución más eficiente, ya que muchas operaciones se pueden ejecutar basándose en los códigos de condición sin necesidad de instrucciones de salto adicionales.

### **10.Pregunta**

**¿Cuáles son las implicaciones de tener diferentes modos de operación en un procesador?**

Respuesta: Los diferentes modos de operación pueden dictar



el nivel de acceso que un programa tiene a los recursos del sistema, con ciertos modos reservados para software a nivel de sistema que requieren controles más estrictos.

## **Capítulo 6 | PARTE CUATRO LA UNIDAD DE CONTROL**

### **Preguntas y respuestas**

#### **1.Pregunta**

**¿Qué papel juega la unidad de control en un procesador y cómo se relaciona con las instrucciones de máquina y las microoperaciones?**

Respuesta:La unidad de control es la parte del procesador responsable de emitir señales de control que gestionan la ejecución de operaciones según lo definido por las instrucciones de máquina. Cada instrucción de máquina en el procesador se descompone en microoperaciones, que son los pasos funcionales más pequeños requeridos para realizar la instrucción. La unidad de control orquesta estas microoperaciones para asegurar que se ejecuten en el orden correcto. Por ejemplo, obtener una



instrucción de la memoria, realizar operaciones aritméticas y lógicas en la ALU, y mover datos entre registros dependen del preciso sincronismo y control gestionado por la unidad de control.

## 2.Pregunta

**¿Cómo se generan y utilizan las señales de control en la ejecución de instrucciones?**

Respuesta: Las señales de control son generadas por la unidad de control basándose en los códigos de operación y flags internos del registro de instrucciones, el estado del reloj y las señales de control de fuentes externas. Estas señales gestionan diversas acciones del procesador, como transferencias de datos entre registros y funciones de la ALU. Por ejemplo, durante el ciclo de recuperación, las señales dirigirán la transferencia del contador de programa al registro de dirección de memoria, permitiendo que el procesador lea la próxima instrucción. El ciclo de ejecución implica diferentes conjuntos de señales según la naturaleza de la operación, como sumar dos números o mover datos de un



registro a otro.

### 3.Pregunta

**¿Puedes explicar el concepto de microoperaciones y su importancia en los ciclos de instrucción?**

Respuesta: Las microoperaciones se refieren a las operaciones elementales que constituyen la ejecución de una instrucción de máquina. Cada ciclo de instrucción, que incluye fases como recuperación, direccionamiento indirecto, ejecución e interrupción, está compuesto por varias microoperaciones.

Por ejemplo, un ciclo de recuperación podría consistir en mover datos del contador de programa al Registro de Dirección de Memoria, luego leer de la memoria al Registro de Búfer de Memoria, y finalmente mover esos datos al Registro de Instrucción. Comprender las microoperaciones es crucial ya que permite a los diseñadores crear señales de control eficientes que gestionen cada paso de las operaciones del procesador sin problemas.

### 4.Pregunta

**¿Cuál es la diferencia entre implementaciones con circuitos fijos y con microprogramación de unidades de**

Más libros gratuitos en Bookey



Escanear para descargar



**control?**

Respuesta: La implementación con circuitos fijos utiliza circuitos lógicos fijos para generar señales de control basándose en la instrucción actual. Este diseño es típicamente más rápido y eficiente, pero menos flexible, ya que cambiar el conjunto de instrucciones requiere modificar el hardware. En contraste, una unidad de control microprogramada define las señales de control a través de un conjunto de microinstrucciones almacenadas en memoria. Esto permite mayor flexibilidad, actualizaciones fáciles y posibles cambios en la operación sin alteraciones físicas al hardware. Sin embargo, puede resultar en un rendimiento más lento debido al tiempo adicional requerido para recuperar microinstrucciones y la complejidad añadida en la decodificación.

## **5.Pregunta**

**¿Cuáles son algunas de las limitaciones y ventajas de usar microprogramación en el diseño de unidades de control?**

Respuesta: Las ventajas de la microprogramación incluyen



mayor flexibilidad y facilidad para actualizar la lógica de control al modificar los microprogramas. Esto hace que sea más sencillo manejar conjuntos de instrucciones complejos y potencialmente permite la adición de nuevas instrucciones sin un rediseño completo del hardware. Las limitaciones incluyen un rendimiento generalmente más lento en comparación con los diseños fijos debido a los ciclos adicionales necesarios para recuperar y decodificar microinstrucciones, y una mayor complejidad en la implementación debido a la necesidad de mecanismos de secuenciación y direccionamiento más sofisticados.





Escanear para descargar



# Por qué Bookey es una aplicación imprescindible para los amantes de los libros



## Contenido de 30min

Cuanto más profunda y clara sea la interpretación que proporcionamos, mejor comprensión tendrás de cada título.



## Formato de texto y audio

Absorbe conocimiento incluso en tiempo fragmentado.



## Preguntas

Comprueba si has dominado lo que acabas de aprender.



## Y más

Múltiples voces y fuentes, Mapa mental, Citas, Clips de ideas...

Prueba gratuita con Bookey



# Capítulo 7 | PARTE CINCO ORGANIZACIÓN PARALELA

## | Preguntas y respuestas

### 1.Pregunta

**¿Cuál es la importancia de la organización paralela en la arquitectura de computadoras?**

Respuesta:La organización paralela permite que múltiples unidades de procesamiento ejecuten aplicaciones de manera cooperativa, mejorando el rendimiento y la disponibilidad. Aborda problemas complejos como el acceso a memoria y el uso compartido de recursos entre procesadores.

### 2.Pregunta

**¿Puedes explicar el problema de coherencia de caché en los sistemas de procesamiento paralelo?**

Respuesta:En sistemas con múltiples cachés, el problema de coherencia de caché surge cuando diferentes cachés tienen copias de los mismos datos. Si una caché actualiza su copia, todas las demás cachés deben ser notificadas para mantener la consistencia de los datos en el sistema.



### 3.Pregunta

**¿Cuáles son las ventajas de los multiprocesadores simétricos (SMP) sobre los sistemas de uniprocador?**

Respuesta:Las ventajas de los SMP incluyen un rendimiento mejorado a través de la ejecución paralela, mayor disponibilidad ya que la falla de un procesador no detiene el sistema, y un crecimiento incremental más fácil al agregar procesadores adicionales.

### 4.Pregunta

**¿Cómo difieren los clústeres de los multiprocesadores simétricos?**

Respuesta:Los clústeres consisten en computadoras independientes que cooperan como un recurso de computación, ofreciendo mejor escalabilidad y mayor disponibilidad en comparación con los SMP, que operan principalmente en memoria compartida.

### 5.Pregunta

**¿Cuáles son las consideraciones de diseño fundamentales para computadoras multicore?**

Respuesta:Las consideraciones clave de diseño para sistemas





multicore incluyen el número de núcleos, los niveles de memoria caché, cómo se comparte la memoria caché y el balanceo de la carga de trabajo entre núcleos para optimizar el procesamiento paralelo.

## 6.Pregunta

**¿Cómo mejoran las técnicas de multihilo el procesamiento multicore?**

Respuesta:El multihilo permite que múltiples hilos se ejecuten de manera concurrente en un procesador multicore, mejorando la utilización general de la CPU y el rendimiento al superponer la ejecución de instrucciones y minimizar el tiempo de inactividad.

## 7.Pregunta

**¿Qué es el protocolo MESI y cuál es su importancia en la coherencia de caché?**

Respuesta:El protocolo MESI es un protocolo de coherencia de caché que mantiene el estado de las líneas de caché en cuatro estados: Modificado, Exclusivo, Compartido e Inválido. Es crítico para garantizar la consistencia entre las



cachés en sistemas multiprocesador.

## 8.Pregunta

**¿Cuál es el papel de la computación vectorial en la computación de alto rendimiento?**

Respuesta:La computación vectorial permite el procesamiento eficiente de grandes arreglos o matrices al realizar operaciones aritméticas simultáneamente en múltiples elementos de datos, lo cual es esencial para tareas en aplicaciones científicas, simulaciones y análisis de datos.

## 9.Pregunta

**¿Cómo difiere el acceso a memoria no uniforme (NUMA) de las arquitecturas tradicionales de memoria compartida?**

Respuesta:NUMA permite que los procesadores accedan a memoria local y remota de una manera en que el tiempo de acceso varía según la ubicación de la memoria. Esto contrasta con las arquitecturas tradicionales donde el tiempo de acceso a memoria es uniforme en todos los procesadores.

## 10.Pregunta

**¿De qué maneras se puede implementar el agrupamiento**





**para mejorar la eficiencia computacional?**

Respuesta:El agrupamiento se puede configurar con discos compartidos o con nodos independientes que trabajan juntos, lo que permite una alta escalabilidad y tolerancia a fallos. La arquitectura se puede expandir de manera incremental, permitiendo que los sistemas crezcan para satisfacer la demanda sin actualizaciones importantes.

## **Capítulo 8 | Apéndice A Proyectos para Enseñar Organización y Arquitectura de Computado**

### **| Preguntas y respuestas**

#### **1.Pregunta**

**¿Por qué se consideran esenciales los proyectos para entender la organización y arquitectura de computado?**

Respuesta:Los proyectos brindan experiencia práctica, permitiendo a los estudiantes interactuar directamente con los conceptos que aprenden.

Ayudan a reforzar el conocimiento teórico al aplicarlo en escenarios prácticos, lo que mejora la



comprensión de las interacciones complejas entre los diferentes componentes de los sistemas de computado.

## 2.Pregunta

**¿Cuáles son algunos tipos de proyectos sugeridos para que los estudiantes trabajen?**

Respuesta: Los tipos de proyectos sugeridos incluyen Simulaciones Interactivas, Proyectos de Investigación, Proyectos de Simulación como SimpleScalar y SMPCCache, Proyectos de Lenguaje de Ensamblado, Asignaciones de Lectura/Informe, y Asignaciones Escritas.

## 3.Pregunta

**¿Cómo mejoran las simulaciones interactivas el aprendizaje en organización y arquitectura de computado?**

Respuesta: Las simulaciones interactivas ayudan a los estudiantes a visualizar mecanismos complejos en los sistemas de computado modernos. Por ejemplo, los estudiantes pueden introducir diferentes parámetros en un simulador de caché para ver cómo cambian las métricas de



rendimiento, promoviendo una comprensión más profunda de los principios de diseño.

#### **4.Pregunta**

**¿Cuál es el propósito de los proyectos de investigación en este contexto?**

Respuesta: Los proyectos de investigación tienen como objetivo reforzar los conceptos básicos del curso mientras enseñan a los estudiantes habilidades de investigación esenciales. Pueden involucrar revisiones de literatura e investigaciones prácticas sobre tecnologías y estándares actuales.

#### **5.Pregunta**

**¿Qué ventajas ofrecen herramientas de simulación como SimpleScalar y SMPCache?**

Respuesta: Estas herramientas permiten modificaciones en los elementos del procesador y características de rendimiento con facilidad, permitiendo a los estudiantes realizar experimentos y analizar los impactos de diversas decisiones arquitectónicas y optimizaciones.



## 6.Pregunta

**¿Cómo facilita CodeBlue el aprendizaje en programación en lenguaje de ensamblado?**

Respuesta:CodeBlue simplifica los conceptos de lenguaje de ensamblado a través de un simulador visual que permite a los estudiantes crear, ejecutar y depurar su código, al mismo tiempo que participan en desafíos de programación competitiva para reforzar su aprendizaje.

## 7.Pregunta

**¿Qué beneficios trae asignar tareas de escritura en cursos técnicos?**

Respuesta:Las asignaciones de escritura fomentan un compromiso más profundo con el material, ayudan a clarificar pensamientos y promueven una comprensión completa de los temas, abordando el problema común de un aprendizaje superficial.

## 8.Pregunta

**¿Cómo funciona el banco de pruebas para evaluar la comprensión?**

Respuesta:El banco de pruebas proporciona una manera



estructurada de evaluar la comprensión del material de los capítulos a través de diversos tipos de preguntas, como verdadero/falso, selección múltiple y completar en blanco, ayudando a los instructores a medir el progreso de los estudiantes.

## 9.Pregunta

**¿Puedes explicar la importancia de los proyectos en la preparación de los estudiantes para aplicaciones del mundo real?**

Respuesta: Los proyectos cierran la brecha entre la teoría y la práctica, equipando a los estudiantes con habilidades prácticas y técnicas de resolución de problemas necesarias para sus futuras carreras en tecnología e ingeniería.

## 10.Pregunta

**¿Cuál es el papel del Centro de Recursos para Instructores (IRC) mencionado en el apéndice?**

Respuesta: El IRC proporciona a los instructores materiales valiosos, incluyendo ideas para proyectos, formatos de asignaciones y recursos para facilitar la enseñanza de conceptos complejos en organización y arquitectura de



computado.

## **Capítulo 9 | Apéndice B Lenguaje de Ensamblaje y Temas Relacionados**

### **| Preguntas y respuestas**

#### **1.Pregunta**

**¿Por qué vale la pena estudiar programación en lenguaje ensamblador?**

**Respuesta:**Estudiar programación en lenguaje

ensamblador ofrece varios beneficios: 1. Aclara la ejecución de instrucciones, ayudándote a entender cómo se ejecuta tu código a nivel de máquina. 2.

Ilustra cómo se almacenan y representan los datos en la memoria, mejorando tu comprensión general de la computación. 3. Demuestra la interacción entre un programa, el sistema operativo, el procesador y los sistemas de E/S, que es vital para la programación de bajo nivel. 4. Ofrece ideas sobre cómo los programas acceden a dispositivos y hardware externos, lo cual es crucial para la



programación de sistemas. 5. Aprender lenguaje ensamblador mejora tus habilidades en lenguajes de alto nivel al darte una mejor comprensión de cómo las construcciones de alto nivel se traducen a instrucciones de máquina.

## 2.Pregunta

**¿Cuáles son los elementos típicos de una instrucción en lenguaje ensamblador?**

Respuesta:Una instrucción típica en lenguaje ensamblador consta de cuatro elementos principales: 1. Etiqueta: Un nombre opcional para una dirección de memoria, utilizado para una referencia fácil. 2. Mnemotécnico: El nombre simbólico de la operación a realizar, que corresponde a las instrucciones de máquina. 3. Operando(s): Una o más variables o direcciones sobre las que actúa la instrucción, que pueden incluir valores inmediatos o referencias a registros o ubicaciones de memoria. 4. Comentario: Texto opcional que describe la funcionalidad de la instrucción para mayor claridad al programador.





### 3.Pregunta

**¿Cuál es la diferencia entre un ensamblador de una pasada y uno de dos pasadas?**

Respuesta:Un ensamblador de una pasada procesa el código fuente en una única pasada, traduciendo las instrucciones a medida que las lee. Esto significa que no puede manejar referencias hacia adelante: etiquetas o símbolos que aún no están definidos en el momento de la traducción. En cambio, un ensamblador de dos pasadas realiza dos pasadas por el código fuente: la primera pasada recopila información sobre etiquetas y sus direcciones, mientras que la segunda pasada traduce las instrucciones a código de máquina utilizando esa información. Esto le permite resolver referencias hacia adelante y producir código ejecutable de manera precisa.

### 4.Pregunta

**¿Cuáles son algunas desventajas de usar lenguaje ensamblador en comparación con lenguajes de alto nivel?**

Respuesta:1. Tiempo de desarrollo: Escribir en ensamblador toma significativamente más tiempo que en un lenguaje de



alto nivel debido a su complejidad y verbosidad. 2. Propenso a errores: El lenguaje ensamblador es menos indulgente; cometer errores puede llevar a bugs graves que son difíciles de depurar. 3. Mantenibilidad: El código ensamblador tiende a ser menos legible y, por lo tanto, más difícil de mantener, especialmente para otros que puedan trabajar en él más tarde. 4. Portabilidad: El código ensamblador es específico de una arquitectura de hardware, lo que dificulta su portabilidad a diferentes sistemas. 5. Funciones limitadas: Carece de construcciones de nivel superior como tipos de datos, técnicas de programación estructurada y soporte de bibliotecas, que son comunes en los lenguajes de alto nivel.

## 5.Pregunta

**¿Qué ventajas tiene el lenguaje ensamblador sobre los lenguajes de alto nivel?**

Respuesta: 1. Rendimiento: El lenguaje ensamblador puede producir código altamente optimizado adaptado a hardware específico, lo que a menudo lleva a una ejecución más rápida. 2. Control: Permite al programador gestionar directamente



los recursos de hardware, ofreciendo un control fino sobre el rendimiento del sistema. 3. Perspectivas de depuración: Entender el ensamblador es útil para depurar lenguajes de alto nivel; examinar el ensamblador generado puede aclarar problemas presentes en el código de alto nivel. 4. Acceso al hardware: Ciertas operaciones e instrucciones pueden no estar disponibles en lenguajes de alto nivel, lo que hace necesario usar lenguaje ensamblador para acceder a ellas. 5. Eficiencia: En entornos con recursos limitados como sistemas embebidos, el ensamblador puede optimizar el uso de memoria y la velocidad de ejecución.

## 6.Pregunta

**¿Puedes describir cómo funcionan los cargadores y los enlazadores en el contexto del lenguaje ensamblador?**

Respuesta:Un cargador es responsable de cargar un programa en la memoria para su ejecución, transfiriendo efectivamente el código ejecutable del disco a la RAM. Configura la imagen del proceso al asignar espacio para el programa, sus datos y la pila de ejecución. Por otro lado, un enlazador



combina varios archivos de objeto creados por el ensamblador en un único archivo ejecutable. Resuelve las referencias entre módulos, ajustando las direcciones en el código objeto para crear un módulo de carga unificado que se puede cargar correctamente en la memoria principal. Esta unificación permite manejar correctamente llamadas a rutinas de biblioteca externas y módulos compartidos, esencial para ejecutar aplicaciones complejas que involucren múltiples componentes.

## 7.Pregunta

**¿Cuál es el propósito de usar una definición de macro en lenguaje ensamblador?**

Respuesta: Las definiciones de macro en lenguaje ensamblador simplifican la escritura de código al permitir que los programadores definan segmentos de código reutilizables o plantillas. Cuando se invoca una macro, se expande en la secuencia completa de instrucciones, reduciendo sustancialmente la duplicación de código y mejorando la mantenibilidad. Esto conduce a una



programación modular, donde secuencias complejas pueden reutilizarse sin la sobrecarga de las llamadas a subrutinas, ya que las macros se expanden en el momento de ensamblar en lugar de en tiempo de ejecución.

## 8.Pregunta

**Da un ejemplo de cómo un programa en ensamblador puede calcular el máximo común divisor (MCD) de dos enteros usando el algoritmo de Euclides.**

Respuesta:Un programa en ensamblador para calcular el MCD puede seguir estos pasos: Comienza cargando dos enteros en registros. Utiliza una estructura de bucle para verificar repetidamente si uno de los números es cero. Si lo es, regresa el otro número como el MCD. Si no, reemplaza el número mayor con el residuo del mayor dividido por el menor. Continúa hasta que el residuo sea cero. El último residuo no cero es el MCD. Por ejemplo, en un lenguaje ensamblador x86, esto podría codificarse aproximadamente de la siguiente manera:



```

start:                ; punto de entrada

test eax, eax        ; verificar si a es 0

jz done              ; si a es 0, saltar a done

mov ebx, edx          ; copiar b a ebx


.loop:                ; inicio del bucle

mov edx, 0            ; limpiar edx antes de la división

div ebx              ; dividir a entre b, cociente en eax, residuo en
edx

mov eax, ebx          ; reemplazar a con b

mov ebx, edx          ; reemplazar b con el residuo

cmp ebx, 0            ; verificar si b es cero

jne .loop             ; repetir si b no es cero


.done:

ret                  ; regresar con el MCD en eax
```


...


```

Esto implementa el cálculo del MCD a través de la aritmética directa y el control de flujo usando conceptos del lenguaje



ensamblador.

**Más libros gratuitos en Bookey**



Escanear para descargar



Ad



Escanear para descargar



App Store  
Selección editorial



22k reseñas de 5 estrellas

## Retroalimentación Positiva

Alondra Navarrete

...itas después de cada resumen  
...en a prueba mi comprensión,  
...cen que el proceso de  
...rtido y atractivo."

**¡Fantástico!**



Me sorprende la variedad de libros e idiomas que soporta Bookey. No es solo una aplicación, es una puerta de acceso al conocimiento global. Además, ganar puntos para la caridad es un gran plus!

Beltrán Fuentes

Fi



Lo  
re  
co  
pr

a Vásquez

hábito de  
e y sus  
o que el  
todos.

**¡Me encanta!**



Bookey me ofrece tiempo para repasar las partes importantes de un libro. También me da una idea suficiente de si debo o no comprar la versión completa del libro. ¡Es fácil de usar!

Darian Rosales

**¡Ahorra tiempo!**



Bookey es mi aplicación de  
crecimiento intelectual. Los  
perspicaces y bellamente c  
acceso a un mundo de con

**¡Aplicación increíble!**



encantan los audiolibros pero no siempre tengo tiempo  
escuchar el libro entero. ¡Bookey me permite obtener  
resumen de los puntos destacados del libro que me  
esa! ¡Qué gran concepto! ¡Muy recomendado!

Elvira Jiménez

**Aplicación hermosa**



Esta aplicación es un salvavidas para los a  
los libros con agendas ocupadas. Los resu  
precisos, y los mapas mentales ayudan a  
que he aprendido. ¡Muy recomendable!

Prueba gratuita con Bookey



# Capítulo 10 | Capítulos en línea| Preguntas y respuestas

## 1.Pregunta

**¿Cuáles son las características clave de los sistemas decimal y binario?**

Respuesta:El sistema decimal utiliza 10 dígitos (0-9) y se basa en potencias de 10, lo que lo hace intuitivo para el uso humano. El sistema binario utiliza solo dos dígitos (0 y 1) y opera en función de potencias de 2, lo cual es esencial para la computación electrónica.

## 2.Pregunta

**¿Cómo se representan los números en el sistema binario?**

Respuesta:En el sistema binario, cada dígito representa una potencia de 2 según su posición. Por ejemplo, el número binario 1011 representa  $(1*2^3) + (0*2^2) + (1*2^1) + (1*2^0) = 8 + 0 + 2 + 1 = 11$  en decimal.

## 3.Pregunta

**¿Qué proceso se utiliza para convertir de binario a decimal?**



Respuesta: Para convertir de binario a decimal, multiplica cada dígito binario por su correspondiente potencia de 2 y suma los resultados. Por ejemplo, para el número binario 1101, calcula  $(1 \cdot 2^3) + (1 \cdot 2^2) + (0 \cdot 2^1) + (1 \cdot 2^0) = 8 + 4 + 0 + 1 = 13$ .

#### 4.Pregunta

**¿Qué es la notación hexadecimal y por qué se utiliza?**

Respuesta: La notación hexadecimal es un sistema numérico de base 16 que utiliza los dígitos 0-9 y las letras A-F. Se utiliza en computación porque representa datos binarios de manera más compacta, facilitando su lectura y escritura para los humanos.

#### 5.Pregunta

**Describe el proceso de convertir fracciones decimales a fracciones binarias.**

Respuesta: Para convertir una fracción decimal a binaria, multiplica la fracción por 2. La parte entera será el dígito binario, y el proceso se repite con la nueva parte fraccionaria hasta que llegue a 0 o se logre la precisión deseada.



## 6.Pregunta

**¿Qué características hacen que la notación hexadecimal sea preferible para los programadores sobre la representación binaria?**

Respuesta:La notación hexadecimal es más compacta y manejable que el binario, que puede ser extenso y engorroso. Esta compactibilidad permite a los programadores trabajar con grandes valores binarios con mayor facilidad y reduce la probabilidad de errores durante la conversión.

## 7.Pregunta

**¿Cómo se relacionan términos clave como radix y base con estos sistemas numéricos?**

Respuesta:El término 'radix' o 'base' se refiere al número de dígitos únicos utilizados en un sistema numérico. Para el decimal, la base es 10; para el binario, es 2; y para el hexadecimal, es 16.

## 8.Pregunta

**¿Por qué es esencial comprender los sistemas numéricos para la organización de computadoras?**

Respuesta:Comprender los sistemas numéricos es crítico para



la organización de computadoras, ya que determinan cómo se representa, procesa y manipula la información en sistemas digitales. Este conocimiento fundamental impacta todo, desde la programación hasta el diseño de hardware.

## 9.Pregunta

**¿Qué desafíos pueden surgir al convertir entre sistemas numéricos?**

Respuesta: Los desafíos incluyen el manejo de posibles ambigüedades, especialmente con fracciones, limitaciones de precisión y asegurar la exactitud durante el proceso de conversión, particularmente con representaciones infinitas en sistemas alternantes.

## 10.Pregunta

**¿Cómo utilizan los programadores estos sistemas numéricos en sus tareas diarias?**

Respuesta: Los programadores convierten frecuentemente entre sistemas numéricos para tareas como la representación de datos en memoria, depuración y optimización del rendimiento en software que involucra programación a bajo



nivel o interacciones con hardware.

**Más libros gratuitos en Bookey**



Escanear para descargar





# Leer, Compartir, Empoderar

Completa tu desafío de lectura, dona libros a los niños africanos.

## El Concepto



×



×



Esta actividad de donación de libros se está llevando a cabo junto con Books For Africa. Lanzamos este proyecto porque compartimos la misma creencia que BFA: Para muchos niños en África, el regalo de libros realmente es un regalo de esperanza.

## La Regla



Gana 100 puntos



Canjea un libro



Dona a África

Tu aprendizaje no solo te brinda conocimiento sino que también te permite ganar puntos para causas benéficas. Por cada 100 puntos que ganes, se donará un libro a África.

Prueba gratuita con Bookey





# Organización y arquitectura de computado Cuestionario y prueba

Ver la respuesta correcta en el sitio web de Bookey

## Capítulo 1 | Organización y arquitectura de computado: Diseñando para el rendimiento (8ª Edición)| Cuestionario y prueba

- 1.El libro 'Organización y arquitectura de computado' fue escrito por William Stallings.
- 2.La octava edición del libro fue publicada por Random House.
- 3.Se requiere permiso para la reproducción o transmisión del contenido del libro.

## Capítulo 2 | Guía para el Lector | Cuestionario y prueba

- 1.El libro 'Organización y arquitectura de computado' de William Stallings está dividido en cinco partes.
- 2.El enfoque principal de la Parte Cuatro del libro es la aritmética computacional y la arquitectura del conjunto de

Más libros gratuitos en Bookey



Escanear para descargar

instrucciones.

3.Los recursos de Internet relacionados con el libro incluyen erratas y enlaces adicionales para estudiantes de informática.

## **Capítulo 3 | VISIÓN GENERAL DE LA PRIMERA PARTE**

### **| Cuestionario y prueba**

- 1.La arquitectura de computadoras se enfoca en las unidades operativas y sus interconexiones.
- 2.La arquitectura ARM se utiliza ampliamente en sistemas embebidos debido a su bajo consumo de energía.
- 3.La ley de Amdahl no está relacionada con las mejoras en el rendimiento del sistema.

**Más libros gratuitos en Bookey**



Escanear para descargar

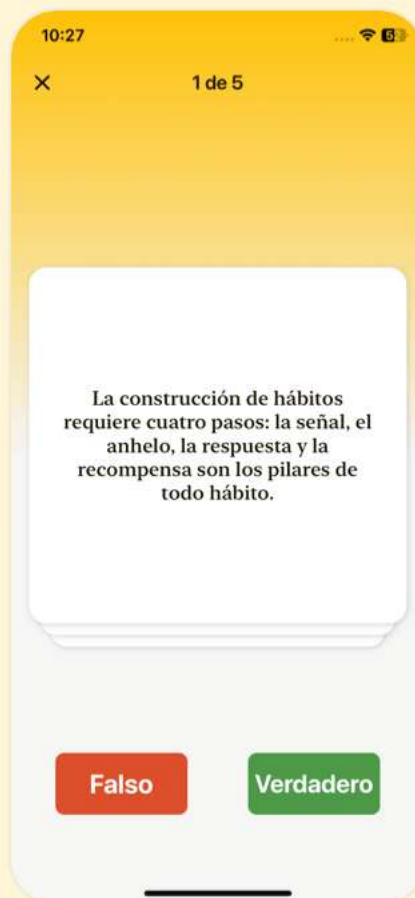


Descarga la app Bookey para disfrutar

# Más de 1000 resúmenes de libros con cuestionarios

**¡Prueba gratuita disponible!**

Escanear para descargar



## **Capítulo 4 | PARTE DOS DEL SISTEMA INFORMÁTICO**

### **Cuestionario y prueba**

- 1.El ciclo de instrucción en un sistema informático incluye las fases de búsqueda y ejecución.
- 2.La memoria caché se utiliza para disminuir las velocidades de acceso al almacenar datos a los que se accede con poca frecuencia.
- 3.La memoria DRAM no requiere ser refrescada debido a fugas de carga.

## **Capítulo 5 | PARTE TRES LA UNIDAD CENTRAL DE PROCESO**

### **| Cuestionario y prueba**

- 1.El capítulo destaca la importancia del pipelining para ejecutar instrucciones de manera eficiente.
- 2.Las arquitecturas CISC utilizan un conjunto de instrucciones más simple en comparación con las arquitecturas RISC.
- 3.La organización de la CPU se discute en términos de sus componentes clave, como los registros y la unidad



aritmética y lógica (ALU).

## **Capítulo 6 | PARTE CUATRO LA UNIDAD DE CONTROL**

### **Cuestionario y prueba**

1. La unidad de control se encarga de secuenciar microoperaciones y generar señales de control para las funciones del procesador.
2. Las unidades de control microprogramadas son menos flexibles que las unidades de control cableadas en la implementación de mecanismos de control.
3. El proceso de ejecución de instrucciones consiste únicamente en los ciclos de búsqueda y ejecución.



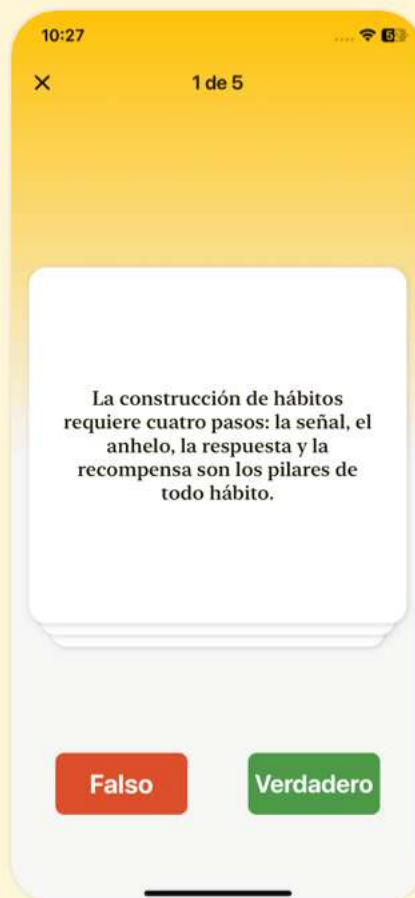


Descarga la app Bookey para disfrutar

# Más de 1000 resúmenes de libros con cuestionarios

**¡Prueba gratuita disponible!**

Escanear para descargar





## **Capítulo 7 | PARTE CINCO ORGANIZACIÓN PARALELA**

### **| Cuestionario y prueba**

- 1.El procesamiento paralelo busca niveles más amplios de paralelismo, permitiendo a múltiples procesadores ejecutar aplicaciones de manera concurrente.
- 2.Los sistemas de Acceso a Memoria No Uniforme (NUMA) tienen tiempos de acceso idénticos para todas las regiones de memoria para todos los procesadores.
- 3.Los sistemas SMP están compuestos por procesadores disímiles que no comparten memoria ni recursos de E/S.

## **Capítulo 8 | Apéndice A Proyectos para Enseñar Organización y Arquitectura de Computado**

### **| Cuestionario y prueba**

- 1.Los proyectos son esenciales para entender los conceptos de organización y arquitectura de computado según muchos instructores.
- 2.El único tipo de proyectos mencionado en el apéndice son





los proyectos de lenguaje ensamblador.

3. Se proporcionan simulaciones interactivas en la edición para ayudar a visualizar mecanismos complejos de sistemas de computado.

## **Capítulo 9 | Apéndice B Lenguaje de Ensamblaje y Temas Relacionados**

### **| Cuestionario y prueba**

1. El lenguaje ensamblador proporciona una representación simbólica del lenguaje de máquina específica para un procesador.
2. El ensamblador de dos pasadas es menos común que el ensamblador de una pasada porque es más complicado.
3. Las declaraciones típicas en lenguaje ensamblador constan de una etiqueta, un mnemónico, operandos y comentarios.





Descarga la app Bookey para disfrutar

# Más de 1000 resúmenes de libros con cuestionarios

**¡Prueba gratuita disponible!**

Escanear para descargar



## Capítulo 10 | Capítulos en línea| Cuestionario y prueba

1. La arquitectura IA-64 utiliza técnicas que se centran en el paralelismo a nivel de instrucción, incluyendo palabras de instrucción very long (VLIW) y predicción de bifurcaciones.
2. La arquitectura IA-64 se basa en la arquitectura x86 tradicional sin cambios significativos.
3. IA-64 soporta la especulación de control para manejar la latencia de memoria al diferir las excepciones hasta que sea necesario.





Descarga la app Bookey para disfrutar

# Más de 1000 resúmenes de libros con cuestionarios

**¡Prueba gratuita disponible!**

Escanear para descargar

