

Algoritmos Paralelos

- Paralelismo como paradigma para resolución de problemas
- Se puede llevar a cabo en aspectos como en simuladores.

▶ Proceso

- Un proceso es cualquier secuencia de operaciones que están ejecutándose en memoria activa
- Permite descubrir el comportamiento de un objeto

▶ Procesos Disjuntos

- Se ejecutan en diferentes bloques de un programa
- No tienen posibilidad de acceder entre sí

▶ Procesos Cooperativos

- Cooperar para realizar una tarea
- Comparten recursos en común
- Requieren alguna forma de sincronización

// Programación Concurrente/Paralelo/distribuido

- Resulta en multiple hilos de control que se comunican entre sí

- resulta en múltiples hilos de control que se comunican entre sí
- Un programa concurrente es aquel por varios procesos que se ejecutan en un solo procesador siguiendo un esquema de paralelismo temporal
- Un programa es aquel formado por varios procesos en varios procesadores conectados mediante una red de comunicación

FALTA

- Un programa distribuido comparte la definición de uno paralelo, la diferencia es que la red de comunicación está en otros lugares geográficos

▶ Ventaja

- Aumento de velocidad
- Eficiencia del procesamiento
- Simplicidad en la expresión de Modelos

▶ Desventajas

- Dificultad de realizar algoritmos en paralelo

≠ ALTA

▶ Organización en la memoria

FALTA

FALTA

▶ Memoria Compartida

FALTA

▶ Memoria Activa

FALTA

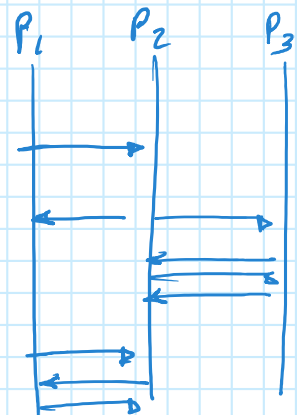
▶ Granularidad

- Es un indicador de la cantidad de trabajo que cada procesador puede realizar de manera independiente.

• FALTA

▶ Granularidad fina

- Los procesadores ejecuten unas cuantas instrucciones y aumentan la cantidad de comunicaciones entre procesadores
- La relación de tiempo



- La relación de tiempo

FALTA

▶ Balance de Carga

- El balance de carga se refiere a la forma de repartir el total de las tareas a realizar entre el total de procesadores disponible
- Balance de carga se divide en 2:
 - Estático
 - Dinámico

▶ Clasificación de Flynn

- Simple Instrucción Simple Datos \rightarrow SISD
- Simple Instrucción Múltiples Datos \rightarrow SIMD
- Múltiple Instrucción Simple Datos \rightarrow MISD
- Múltiple Instrucción Múltiples Datos \rightarrow MIMD

TAREA | Describir cada uno e indicar 2 ejemplos de cada uno

▶ Clasificación Algorítmica

1] Paralelismo Algorítmico

- Se basa en la paralelización de tareas cuasi-independientes que se ejecutan secciones del algoritmo
- Se relaciona con el modelo de Flujo de datos ya que

FALTA

2] Paralelismo Geométrico

- Se basa en tareas independientes
- Consiste en dividir los datos del problema

se basa en tareas independientes

- Consiste en dividir los datos del problema de una manera simétrica considerando una distribución uniforme, siendo cada proceso

3] Paralelismo Farm (Manager Worker)

- Debido a la independencia, se necesita de un coordinador que se encarga de distribuir el trabajo

▶ Métricas de Desempeño

1] Tiempos de procesamiento y tiempo comunicación

- El tiempo puede subdividirse entre el tiempo

~~FALTA~~

2] SpeedUp

3] Eficiencia

La medida de eficiencia se asocia a la idea

La medida de eficiencia se asocia a la idea

4] Fracción Serial

► Niveles de Paralelismo

► Paralelismo a nivel de instrucción

- Tareas independientes se pueden ejecutar en diferentes procesadores

• FALTA

Tema: Archivos

- Compromiso | Memoria Más Cara → Menor cantidad
Espacio/Tiempo | ↳ Más veloz
- Los archivos son permanentes en memoria secundaria (Tiempo indefinido)
- Un archivo es un flujo de bytes de información
- Operaciones |
 - Eliminar
 - Editar
 - Crear

Pregunta
Examen

- ① Principal forma de almacenamiento de información
- ② Los sistemas operativos los utilizan como primitivas

Organización Física

- Cada plato puede ser leído o escrito por una cabeza (head) que se encuentra al final del brazo (arm)

Movimientos |

- Brazo
- Discos

Mecánicos

- Paginación y segmentación de los discos

- Paginación y segmentación de los discos (Lo hace el OS)

► Organización Lógica

- 1 byte = 8 bits
- 1 kilobyte = 1024 bytes
- 1 megabyte = 1048576 = (2^{20}) bytes
- 1 gigabyte = 1024 megabyte = 2^{30} bytes

→ Memoria Principal

→ Memoria Externa *10 mil veces más lento*

- Se consideran bloque para el uso y movimientos de estos,

$$\bullet \text{ N\u00fam Bloques Disponibles} = \frac{\text{Tam. memoria}^*}{\text{Tam. del bloque}}$$

$$\bullet \text{ Cantidad de registros} = \frac{\text{Tamaño del bloque}}{\text{Tamaño de cada registro}}$$

$$\bullet \text{ N\u00fam de bloques necesarios} = \frac{\text{Cantidad de registros}}{\text{Cantidad de registros}}$$

bloques
necesarios

Cantidad de registros
por bloque

➔ Implementación Archivos

FALTA