

**Universidad Autónoma del Estado de México**

**Unidad Académica Profesional Tianguistenco**

**Ingeniería en software**

**Unidad de aprendizaje:**

Programación Paralela

**Profesor:**

Ing. Gustavo

**Título:**

**Programación Paralela**

**Alumno:**

Armando Jovani Martínez Ruiz

La computación paralela es el uso de múltiples recursos computacionales para resolver un problema. Se distingue de la computación secuencial en que varias operaciones pueden ocurrir simultáneamente.

El paralelismo clásico, o puesto de otra manera, el clásico uso del paralelismo, es el de diseño de programas eficientes en el ámbito científico. La simulación de problemas científicos es un área de gran importancia, los cuales requieren de una gran capacidad de procesamiento y de espacio de memoria, debido a las complejas operaciones que se deben realizar.

**Características:**

* Un problema se divide en partes discretas que se pueden resolver simultáneamente
* Cada parte se descompone en una serie de instrucciones
* Las instrucciones de cada parte se ejecutan simultáneamente en diferentes procesadores
* Se emplea un mecanismo global de control/coordinación

**Ventajas**

* Resuelve problemas que no se podrían realizar en una sola CPU
* Resuelve problemas que no se pueden resolver en un tiempo razonable
* Permite ejecutar problemas de un orden y complejidad mayor
* Permite ejecutar código de manera más rápida (aceleración)
* Permite ejecutar en general más problemas
* Obtención de resultados en menos tiempo
* Permite la ejecución de varias instrucciones en simultáneo
* Permite dividir una tarea en partes independientes

**Desventajas**

* Mayor consumo de energía
* Mayor dificultad a la hora de escribir programas
* Dificultad para lograr una buena sincronización y comunicación entre las tareas
* Retardos ocasionados por comunicación ente tareas
* Número de componentes usados es directamente proporcional a los fallos potenciales
* Condiciones de carrera
* Múltiples procesos se encuentran en condición de carrera si el resultado de los mismos depende del orden de su llegada
* Si los procesos que están en condición de carrera no son correctamente sincronizados, puede producirse una corrupción de datos

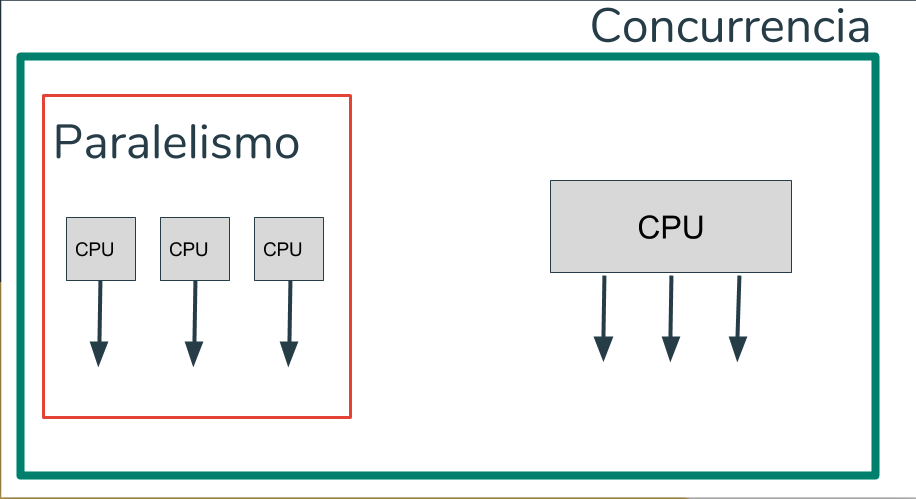
**Concurrencia vs Paralelismo**

**Concurrencia**

Capacidad de operar actividades al mismo tiempo. Es decir se pueden tener varios procesos corriendo cada uno en un procesador o puede haber varios proceso que corran solo en un procesador

**Paralelismo**

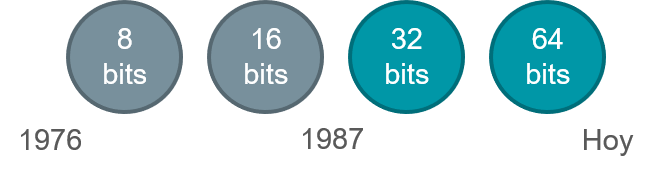
Son muchas actividades teniendo lugar al mismo tiempo, “la cualidad o el estado de ser paralelo”. El hecho de ser paralelo implica que solo se pueden tener varios procesos corriendo cada uno en un procesador.



**Tipos de paralelismo**

* **Paralelismo a nivel de bit:**

Se habla de paralelismo al nivel de bit, cuando se **aumenta el tamaño de la palabra del procesador** (tamaño de la cadena de bits a procesar). Este aumento reduce el número de instrucciones que tiene que ejecutar el procesador en variables cuyos tamaños sean mayores a la longitud de la cadena.  
**Ejemplo:** En un procesador de 8-bits sumar dos números de 16bits tomaría dos instrucciones. En un procesador de 16-bits esa operación requiere solo una instrucción.

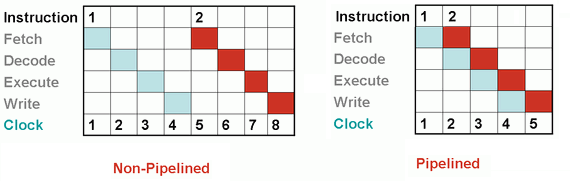


* **Paralelismo a nivel de instrucción**

Este tipo de paralelismo consiste en **cambiar el orden de las intrucciones** de un programa y juntarlas en grupos para posteriormente ser ejecutados en paralelo **sin alterar el resultado final** del programa.

**Pipelining**

El pipelining proviene de la idea de que en una tubería no es necesario esperar a que toda el agua dentro salga, para que pueda entrar más.

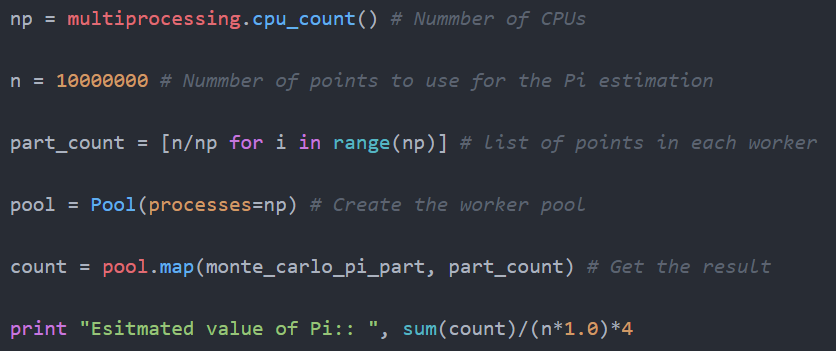


* **Paralelismo a nivel de datos**

Cada procesador realiza la misma tarea sobre un subconjunto independiente de datos.

Ej: Dos granjeros se dividen el área de césped a podar.

El caso clásico de paralelismo de datos, es el cálculo de pi por partes usando el método de monte carlo:



* **Paralelismo a nivel de tareas**

**Cada hilo** realiza **una tarea distinta** e independiente de las demás.  
**Ej:** Un granjero poda el césped, el otro cosecha.

