

14. Según la taxonomía de Flynn clasifique OPENMP, MPI y Multiprocessing. Justifique su respuesta.

La taxonomía de Flynn clasifica los sistemas informáticos según la cantidad de instrucciones y datos que se pueden procesar simultáneamente, y distingue entre instrucciones únicas o múltiples y datos únicos o múltiples. Según esta clasificación:

1. OpenMP (Open Multi-Processing)

Es una API que soporta programación multi-hilo en arquitecturas de memoria compartida. Permite la ejecución paralela de código mediante la creación de múltiples hilos que comparten el mismo espacio de memoria.

Clasificación: **MIMD (Multiple Instruction, Multiple Data)**

En OpenMP, múltiples hilos pueden ejecutar diferentes instrucciones (en diferentes partes del código) y operar sobre diferentes datos de manera simultánea. Esto se alinea con la definición de MIMD, donde cada hilo (unidad de control) puede tener su propia secuencia de instrucciones y datos.

2. MPI (Message Passing Interface):

MPI es una especificación de biblioteca para la programación de paso de mensajes, comúnmente utilizada en sistemas distribuidos y paralelos.

Clasificación: **MIMD (Multiple Instruction, Multiple Data)**

Ya que múltiples procesadores ejecutan múltiples instrucciones en diferentes flujos de control y pueden operar sobre múltiples conjuntos de datos al mismo tiempo. Esto se debe a que en MPI, los procesos pueden ejecutar diferentes partes del programa y comunicarse entre sí mediante el intercambio explícito de mensajes, lo que permite un alto grado de paralelismo y distribución de datos.

3. Multiprocessing:

El multiprocessing permite la creación y gestión de procesos paralelos

Clasificación: **MIMD (Multiple Instruction, Multiple Data)**

multiprocessing utiliza procesos del sistema operativo en lugar de hilos. Esto permite aprovechar mejor los múltiples núcleos de la CPU, ya que cada proceso puede ser programado en un núcleo diferente, ejecutándose en paralelo de manera real.

Cada proceso en multiprocessing tiene su propio espacio de direcciones de memoria, lo que garantiza que las operaciones de un proceso no interfieran directamente con las de otro. Esto es característico de un entorno MIMD donde cada unidad de procesamiento (en este caso, cada proceso) trabaja independientemente en su propio conjunto de instrucciones y datos.

En multiprocessing, diferentes procesos pueden realizar tareas completamente diferentes. Por ejemplo, un proceso puede estar leyendo datos de un archivo mientras otro está procesando datos y un tercero está enviando resultados a través de una red. Esta capacidad de ejecutar diferentes flujos de instrucciones simultáneamente sobre diferentes conjuntos de datos es una característica distintiva del MIMD, cada procesador ejecuta instrucciones independientes en flujos de control separados y puede trabajar con diferentes conjuntos de datos simultáneamente.