

La taxonomía de Flynn clasifica las arquitecturas de computadoras según su capacidad para procesar instrucciones y datos de manera simultánea, estableciendo una relación directa con la programación paralela al determinar qué tipo de arquitectura es más adecuada para una aplicación específica.

### **OpenMP:**

Es una API para la programación paralela en sistemas de memoria compartida que utiliza directivas para la ejecución paralela de código, como `#pragma omp parallel` para ejecutar código en paralelo

### **Multiprocessing (Python):**

Esta librería se utiliza para crear y gestionar procesos, ofreciendo la posibilidad de ejecución paralela tanto en un mismo ordenador como entre ordenadores remotos.

### **NumPy:**

Es una librería de Python enfocada en el cálculo numérico que ofrece estructuras de datos como arreglos y matrices, optimizadas para operaciones matemáticas.

### **mp (Python):**

Proporciona funcionalidades para la creación y gestión de procesos en paralelo. Las clases como `mp.Process`, `mp.Pool`, `mp.Queue` y `mp.Pipe` son utilizadas para procesos paralelos y comunicación entre ellos.

### **Threading (Python):**

Facilita la creación y ejecución de hilos (threads), que son secuencias de ejecución dentro de un proceso mayor, permitiendo la paralelización a nivel de hilo.

### **C# Threading:**

En C#, la paralelización se puede lograr a través de hilos con la clase `System.Threading.Thread`, que proporciona control sobre la ejecución de hilos.