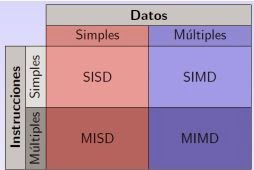
**14.- Según la taxonomía de Flynn clasifique OPENMP, MPI y Multiprocessing . Justifique su respuesta.**

La taxonomía de Flynn clasifica los sistemas informáticos según la forma en que manejan las instrucciones y los datos. Tiene cuatro categorías principales:

* SISD (Single Instruction, Single Data): Un solo flujo de instrucciones que opera en un solo flujo de datos. Es el modelo tradicional de computación secuencial.
* SIMD (Single Instruction, Multiple Data): Una sola instrucción opera simultáneamente en múltiples conjuntos de datos. Es común en arquitecturas de procesadores vectoriales.
* MISD (Multiple Instruction, Single Data): Múltiples flujos de instrucciones operan en un solo flujo de datos. Es una categoría teórica y poco común en la práctica.
* MIMD (Multiple Instruction, Multiple Data): Múltiples flujos de instrucciones operan simultáneamente en múltiples flujos de datos. Es el tipo de sistema más común para la computación paralela.



Ahora, vamos a clasificar OPENMP, MPI y Multiprocessing según la taxonomía de Flynn:

* OPENMP (Open Multi-Processing): OPENMP es una API de programación para la programación multiproceso en sistemas de memoria compartida. Utiliza una aproximación MIMD, ya que múltiples instrucciones pueden ser ejecutadas en diferentes hilos de ejecución, pero estos hilos comparten la misma memoria. Por lo tanto, OPENMP se clasifica dentro de la categoría MIMD.
* MPI (Message Passing Interface): MPI es una especificación para la programación de sistemas distribuidos, donde múltiples procesos se ejecutan en diferentes nodos de un clúster y se comunican entre sí mediante mensajes. MPI también se clasifica como MIMD, ya que permite la ejecución de múltiples instrucciones en múltiples datos, aunque estos datos se encuentran distribuidos en diferentes nodos.
* Multiprocessing: EL multiprocesamiento es una técnica que implica el uso de múltiples unidades de procesamiento dentro de un único ordenador o en un clúster. Dependiendo de cómo se implemente, puede ser clasificado como MIMD si los procesos se ejecutan de manera independiente, o como SIMD si los procesos comparten la misma instrucción, pero operan en diferentes conjuntos de datos.