**EJERCICIO 2**

1. **OpenMP y MPI de cual elemento taxonómico anterior es.**

**OpenMP**

es una [interfaz de programación de aplicaciones](https://es.wikipedia.org/wiki/Interfaz_de_programaci%C3%B3n_de_aplicaciones) (API) para la programación [multiproceso](https://es.wikipedia.org/wiki/Multiproceso) de [memoria compartida](https://es.wikipedia.org/wiki/Memoria_compartida) en múltiples plataformas. Permite añadir [concurrencia](https://es.wikipedia.org/wiki/Computaci%C3%B3n_concurrente) a los programas escritos en C, C++ y Fortran sobre la base del modelo de ejecución [fork-join](https://es.wikipedia.org/wiki/Bifurcaci%C3%B3n_(sistema_operativo)" \o "Bifurcación (sistema operativo)). Está disponible en muchas [arquitecturas](https://es.wikipedia.org/wiki/Arquitectura_de_computadoras), incluidas las plataformas de Unix y de Microsoft Windows. Se compone de un conjunto de [directivas de compilador](https://es.wikipedia.org/wiki/Pragma#Programaci%C3%B3n), rutinas de biblioteca, y [variables de entorno](https://es.wikipedia.org/wiki/Variable_de_entorno) que influyen el comportamiento en tiempo de ejecución.

Definido conjuntamente por proveedores de hardware y de software, OpenMP es un modelo de programación [portable](https://es.wikipedia.org/wiki/Portabilidad) y [escalable](https://es.wikipedia.org/wiki/Escalabilidad) que proporciona a los programadores una interfaz simple y flexible para el desarrollo de aplicaciones paralelas, para plataformas que van desde las computadoras de escritorio hasta [supercomputadoras](https://es.wikipedia.org/wiki/Supercomputadora). Una aplicación construida con un modelo de [programación paralela](https://es.wikipedia.org/wiki/Computaci%C3%B3n_paralela) híbrido se puede ejecutar en un [cluster de computadoras](https://es.wikipedia.org/wiki/Cluster_(inform%C3%A1tica)" \o "Cluster (informática)) utilizando OpenMP y [MPI](https://es.wikipedia.org/wiki/Interfaz_de_Paso_de_Mensajes), o a través de las extensiones de OpenMP para los sistemas de [memoria distribuida](https://es.wikipedia.org/w/index.php?title=Memoria_distribuida&action=edit&redlink=1).

**Respuesta: entonces podemos decir que openMp es elemento de MISD**

**MPI**

[**MPI**](https://lsi2.ugr.es/jmantas/ppr/ayuda/mpi_ayuda.php) ("Message Passing Interface", Interfaz de Paso de Mensajes) es un estándar que define la sintaxis y la semántica de las funciones contenidas en una biblioteca de paso de mensajes diseñada para ser usada en programas que explotan la existencia de múltiples procesadores.  
El paso de mensajes es una técnica empleada en programación concurrente para aportar sincronización y comunicación entre procesos de manera similar a como se hace con los semáforos, monitores, etc.  
Su principal característica es que no precisa de memoria compartida, por lo que es muy importante en la programación de sistemas distribuidos.

como es un interfaz estandarizado para la realización de aplicaciones paralelas basadas en pasaje de mensajes. El modelo de programación que subyace tras MPI es MIMD (Multiple Instruction streams, Multiple Data streams) aunque se dan especiales facilidades para la utilización del modelo SPMD (Single Program Multiple Data), un caso particular de MIMD en el que todos los procesos ejecutan el mismo programa, aunque no necesariamente la misma instrucción al mismo tiempo. El pasaje de mensajes es un paradigma ampliamente usado en determinadas clases de máquinas paralelas, especialmente aquellas con memoria distribuida. Aun habiendo bastantes variaciones, los conceptos básicos de las comunicaciones entre procesos por medio de mensajes están bien estudiados, en los últimos 15 años se han hecho progresos sustanciales en lo que hace a las aplicaciones de este paradigma y cada distribuidor ha implementado su propia variante, así es como recientemente muchos sistemas han demostrado ser eficientes y portables. De esta manera y con el tiempo es como se ha definido la sintaxis y la semántica de un núcleo de librerías de rutinas que hace del pasaje de mensajes una herramienta absolutamente útil para un amplio rango de usuarios e implementable sobre variedad de computadoras.

**Respuesta: Se pude decir que MPI es MIMD aunque en dados casos se puede dar una mayor facilidad en el uso de SPMD**

**Entonces en conclusión openMP usa el elemento MISD y MPI es elemento MIMD**