

1. Describa de manera teórica los siguientes conceptos: SISD, SIMD, MISD y MIMD. Indique además que lenguajes aplican a estos.

R1:

-SISD (Single Instruction Single Data): Traducido al español significa que se ejecuta un solo flujo de instrucciones en un solo flujo de datos. Ejemplo: Los lenguajes de programación conocidos (C, C++, Python...) pueden seguir esta arquitectura modelo con programas secuenciales muy simples.

-SIMD (Single Instruction Multiple Data): Traducido al español significa que se ejecuta un solo flujo de instrucciones en varios flujos de datos. Por ejemplo si observamos el fragmento del siguiente programa utilizando la librería OPENMP en C++ estamos siguiendo esta arquitectura modelo:

```
// definimos 2 flujos de datos:
int flujoDeDatos1[2]; memset(flujoDeDatos1, 0, sizeof flujoDeDatos1);
int flujoDeDatos2[2]; memset(flujoDeDatos2, 0, sizeof flujoDeDatos2);
// definimos 1 solo flujo de instruccion (cada hilo hace lo mismo):
#pragma omp parallel
{
    for (int i=1; i<=5; i++){
        flujoDeDatos1[0]+=3*i + i*i*i;
        flujoDeDatos1[1]+=-7 + 6*i + i*i;

        flujoDeDatos2[0]+=i*i*i;
        flujoDeDatos2[1]+=-100 +8*i;
    }
}
```

-MISD (Multiple Instruction Single Data): Traducido al español significa que se ejecutan múltiples flujos de instrucciones en un solo flujo de datos. Por ejemplo si observamos el fragmento del siguiente programa utilizando la librería OPENMP en C++ estamos siguiendo esta arquitectura modelo:

```
// definimos un solo flujo de dato:
int suma=0;
// definimos una instrucción diferente cuando el hilo es par o impar:
#pragma omp parallel
{
    int hilo=omp_get_thread_num();
    if (hilo&1){
        suma+=300;
    }else{
        suma+=130;
    }
}
```

-MIMD (Multiple Instruction Multiple Data): Traducido al español significa que se ejecutan varios flujos de instrucciones en múltiples flujos de datos. Por ejemplo en mi solución del ejercicio 14 (con la librería MPI de C++) de este examen se puede observar que se usa esta arquitectura modelo. Porque tenemos varios flujos de datos definidos

(por ejemplo: las matrices 1 y 2 instanciadas desde el procesador Master se puede definir como el Flujo de Datos #1, y los arreglos auxiliares "row" y "col" que utilizan los procesadores Slave se definen como el Flujo de Datos #2) y también tenemos varios flujos de instrucciones (porque lo que hace el procesador Master no es lo mismo que hacen los procesadores Slave, ya que el Master distribuye el trabajo y los Slave hacen el trabajo).

En conclusión, se pueden modelar estas arquitecturas con diferentes lenguajes de programación (por ejemplo podríamos utilizar la librería "multiprocessing" de python).