

Laboratorio 5: Explotando el paralelismo con Python en CPUs

El apartado preliminar de este laboratorio fue sencillo preparando el repositorio github y la conexión al JupyterNotebook.

La práctica plantea una operación sencilla de entender, pero que requiere gran potencia de cálculo, así en el apartado 3.2 me familiaricé con los tiempos de ejecución del código plano y aprendí a aplicar el operador `njit` dentro de `numba` para mejorar la velocidad de ejecución. Seguidamente la aplicación del paquete `multiprocessing` con `Pool` requirió algo de bibliografía para ver cómo se utilizaba el paquete, pero adecuadamente lo pude implementar. Por último, `numba` con `prange` permitió mejorar aún más el tiempo de ejecución sin complicar el código excesivamente. Este apartado fue muy visual al ver una reducción drástica de los tiempos de ejecución al aplicar los distintos métodos e implementando el paralelismo en la ejecución.

En el apartado 3.3 la implementación del código también fue sencilla al estar ya familiarizado con los paquetes necesarios y al ser un proceso más costoso se vió mejor la optimización al paralelizar los procesos con 1,2,4 y 8 núcleos. Además, los valores introducidos fueron tan grandes que dejaron patente la diferencia en velocidad de cálculo de cálculo de cada celda de código.

Por último, la actividad del cálculo de π resultó muy interesante al aplicar los paquetes aprendidos en algo palpable y con un experimento conocido como el método MonteCarlo para el cálculo de dicho número.

Este laboratorio no siendo excesivamente complicado ayudó a mi aprendizaje de los distintos paquetes.