

Informe del lab 5_Python_Advanced

Antonio José Manuel Martínez

Esta práctica ha permitido trabajar con distintas técnicas de optimización y paralelización en Python, aplicadas al cálculo de números primos y a la estimación de π mediante Monte Carlo. El enfoque progresivo, desde la versión secuencial hasta las optimizaciones con *NumPy*, *Numba* y *multiprocessing*, ha sido uno de los aspectos más positivos. Se aprecia claramente cómo cada método elimina diferentes cuellos de botella y cómo el uso de herramientas como *Numba* puede transformar de forma notable el rendimiento de programas intensivos. También ha sido especialmente útil ejecutar los experimentos en el clúster, ya que trabajar con colas, recursos y scripts SLURM aporta una experiencia práctica muy cercana al uso real de sistemas HPC.

Otro aspecto destacable es que la práctica ayuda a comprender las diferencias entre trabajar con procesos, paralelismo automático y vectorización. Además, obligar a integrar GitHub y mantener el repositorio actualizado fomenta hábitos de trabajo reproducibles y ordenados.

Entre los aspectos mejorables, en algunos puntos se echa en falta una explicación más clara sobre el comportamiento interno de Numba en modo paralelo y sobre cómo gestionar correctamente la aleatoriedad en procesos concurrentes. También genera cierta confusión que algunas colas del clúster no estaban accesibles al principio (la cola Mendel), lo que me provocaba errores. Por suerte ese problema se solucionó con celeridad.

En conjunto, la práctica ha sido muy completa y útil para entender cómo optimizar y paralelizar código en Python dentro de un entorno HPC. La combinación de técnicas vistas, junto con la ejecución real en el clúster, proporciona una visión clara y aplicable al trabajo futuro tanto académico como profesional.