

## Informe práctica 5: Paralelismo (núcleos y GPUs) con Python en GACOP

Marta Campillo Lizán

Con esta práctica he aprendido a optimizar procesos en Python, pero sobre todo me ha servido para machacar conceptos que ya habíamos manejado en las tareas anteriores, ya que la paralelización en sí es entender cómo se estructuran los procesos y aplicar los comandos propios de los paquetes. Creo que en cuanto al aspecto de la optimización, es de las prácticas que podríamos aplicar realmente fuera de la asignatura.

En general no la he encontrado muy enrevesada, sin embargo me gustaría señalar que he perdido un poco de tiempo usando la cola bohr al intentar ceñirme al enunciado. Al rato he tenido que pasarme a la de nikola-gpu porque en bohr no me reconocía los comandos de cupy.

En cuanto al ejercicio extra, no he podido implementar el punto del decorador @reduce de Numba.

### Usando el decorador @reduce de Numba

```
[60]: from numba import reduce

@reduce
def add_reduce(a, b):
    return a + b

-----
ImportError                                Traceback (most recent call last)
Cell In[60], line 1
----> 1 from numba import reduce
      3 @reduce
      4 def add_reduce(a, b):
      5     return a + b

ImportError: cannot import name 'reduce' from 'numba' (/usr/local/lib64/python3.8/site-packages/numba/__init__.py)
```