

# Informe sobre exploración y simulación de procesos paralelos y concurrentes.

## Integrantes:

Juan Manuel Perea Coronado - 1926462.

Andrés Mauricio Ramirez - 1926987.

## Introducción.

Esta actividad se desarrolló en Python, considerando las características del equipo dónde se ejecutó (12 núcleos). En ella se muestran las diferencias en los tiempos de ejecución de cada uno de los procesos, según cómo fueron ejecutados: secuencial, concurrencia o paralelismo.

## Desarrollo de la actividad.

Se ejecutaron 5 veces los códigos para la ejecución de los procesos de forma secuencial (secuencial.py), concurrente (FiboWorkerThread.py) y paralela (FiboWorkerProcess.py) y el resultado de los tiempos en cada ejecución es el siguiente:

Número de Ejecución.	Tiempo en Segundos.		
	Secuencial	Concurrente	Paralelo
Primera	67.4082	73.8394	22.5921
Segunda	65.1430	68.8346	21.8515
Tercera	64.7556	70.7647	22.1665
Cuarta	68.0896	71.6362	22.9136
Quinta	64.5358	69.2581	23.4970

**Tabla 1. Tiempos de la ejecución del proceso de diferentes formas.**

Eliminando los valores extremos de cada tipo de ejecución y promediando, se tiene el siguiente resultado:

Tipo de Ejecución.	Promedio de Tiempo en Segundos.
Secuencial	65.7689

Concurrente	70.553
Paralelo	22.5574

**Tabla 2. Tiempos promedio de la ejecución del proceso de diferentes formas.**

## **Conclusión.**

De lo anterior, se puede observar que el ejecutar el proceso de forma paralela, el tiempo promedio que demora en terminarlo es mucho menor que de las otras maneras, siendo hasta tres veces más rápido. También se tiene que, en promedio, el tiempo de ejecución y de forma concurrente son muy similares, pero este último es al que le toma más tiempo terminar.

## **Anexos.**

Los códigos usados están disponibles en el siguiente repositorio de forma pública:  
<https://github.com/Juanmaperea/Secuencial-Hilos-Procesos.git>