***Ingeniería de Sistemas y Computación***

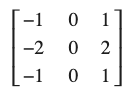
***Reinel Tabares Soto - Johan S. Piña - Harold Brayan Arteaga A.***

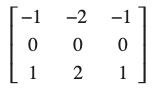
***Proyecto Final***

***Programación Concurrente y Distribuida***

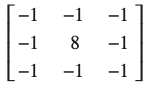
Como proyecto final del curso se solicita los siguientes ítems:

1. Cargar automáticamente en memoria 10,000 imágenes aleatorias tomadas del buscador de google de un tema específico, para este caso se debe implementar 10 hilos para que cada uno descargue 1,000 imágenes de forma concurrente. [Recurso](https://github.com/rramosp/20182.mldl/blob/master/contents/SESION%205.4%20-%20Construccion%20de%20datasets%20de%20imagenes.ipynb)
2. Realizar filtrado de imágenes según los siguientes kernels de la Figura 3 del paper *Sensitivity of deep learning applied to spatial image steganalysis* [LINK](https://doi.org/10.7717/peerj-cs.616):
   1. Kernel “El primero de los Class 1”
   2. Kernel “El primero de los Class 2”
   3. Kernel “El primero de los Class 3”
   4. Kernel “Square 3x3”
   5. Kernel “El primero de los Edge 3x3”
   6. Kernel “Square 5x5”
   7. Kernel “El primero de los Edge 5x5”
   8. Kernel de sobel vertical y horizontalmente

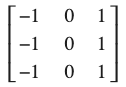


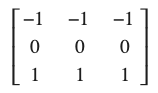


* 1. Kernel de Laplace



* 1. Kernel de prewitt vertical y horizontal





Se debe mostrar las dimensiones de la imagen luego de filtrada, valor mínimo, valor máximo, valor medio y desviación estándar de los píxeles.

1. El filtrado se debe poder realizar en los siguientes lenguajes de programación, frameworks o librerías:
   1. C
   2. OpenMP
   3. Multiprocessing
   4. MPI4PY
   5. PyCUDA
2. Se debe diseñar una interfaz en streamlit que permita al usuario elegir el tipo de filtro, el tipo de framework o librería y la forma de distribuir el proceso en el caso de que sea paralelo (cuantos cores, procesos o hilos se van a utilizar, según disponibilidad del sistema donde se ejecute la app). [Recurso 1](https://drive.google.com/file/d/1N6kveB0pffZ7hWkdedJokdPD98Frswes/view)
3. El sistema debe mostrar 10 imágenes aleatorias con los resultados, los tiempos de duración en serie y en paralelo según los recursos elegidos en la interfaz, su aceleración y un breakdown de tiempos.
4. Generar un informe tipo paper en [Overleaf](https://www.overleaf.com/latex/templates/formato-articulo-cieta/hcyrpjrzxmtk) donde se tengan los siguientes puntos:
   1. Introducción.
   2. Metodología.
   3. Resultados.
   4. Discusión.
   5. Conclusiones.
   6. Referencias.
5. El proyecto es en parejas.

**Entregables:**

* **Documento tipo paper en un PDF**: Según las instrucciones dadas.
* **Enlace a la Página Web de Streamlit**: La página web debe ser accesible y mostrar claramente cómo interactuar con ella.
* **Enlace al Repositorio en GitHub**: Se espera que el código fuente del proyecto esté disponible en un repositorio público en GitHub. Este repositorio debe incluir el código de la implementación de la interfaz en Streamlit, y cualquier otro archivo relevante para la replicación y comprensión del proyecto.
* **Enlace a un Video en YouTube** (Oculto, Visible Solo con el Enlace): Presente un video conciso de menos de 10 minutos que destaque las funcionalidades clave del proyecto. Se debe dar una demostración de funcionamiento. Los dos estudiantes exponen o explican en equipo el proyecto.

**Fecha de entrega: 10 de diciembre de 2023**

**¡Muchos éxitos!**