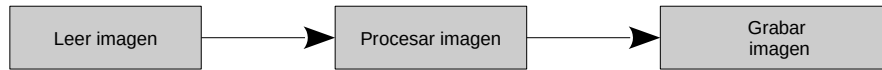


Proyecto Parcial SO

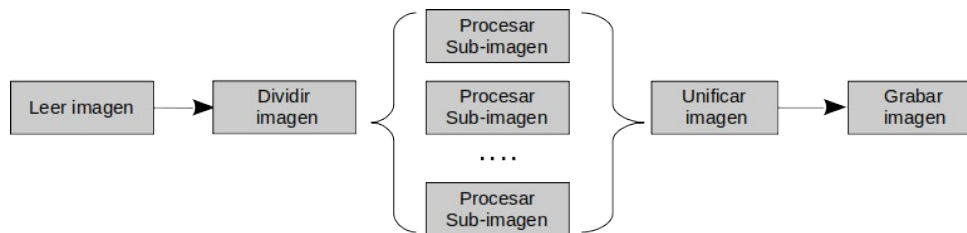
Convolución concurrente

1. Descripción del problema

www.DarasViendo.com es una startup ecuatoriana que quiere implementar un "pipeline" de procesamiento de imágenes para crear un sistema de IA para realizar búsqueda de imágenes de forma automática similar a Google. Un pipeline básico tiene 3 pasos: leer, procesar y escribir imágenes.



Para reducir el tiempo de procesamiento, le plantean a Ud. como experto en sistemas operativos que diseñe e codifique un pipeline que procese imágenes de forma concurrente y/o paralela.



Su diseño deben considerar los siguientes requerimientos mínimos:

- La lectura y escritura de una imagen debe ser asíncrona (es decir puede ocurrir en cualquier momento)
- El procesamiento de imágenes se limitará a la operación de convolución usando filtros lineales. Cada imagen puede ser filtrarse con alguno de estos filtros de: sobel, blur, sharpen de tamaño 3x3.
- La imagen debe dividirse de forma que N – 1 CPUs procesen las sub-imágenes de forma paralela en el menor tiempo posible.
- Una vez que las sub-imágenes hayan sido procesadas las imágenes resultantes debe almacenarse en disco duro de forma secuencial usando un esquema FIFO.
- Mientras se procesa una imagen es posible que llegue otro requerimiento (el nombre de archivo de otra imagen para ser procesada, junto con el filtro a utilizar).

2. Entregables (en un zip)

1. Código fuente escrito en C (ANSI), C++.
2. Archivo Readme.txt con instrucciones de compilación y ejemplo de tests.
3. Documento de diseño (máximo 3 páginas) PDF, un grafico.

3. Recursos Se provee una librería para lectura y escritura de imágenes en formato PGM.

4. Rúbrica

Item	Ponderación
Los mecanismos de multiprogramación seleccionados cumplen con los requerimientos de forma <u>EFICIENTE</u>	20
Los mecanismos de IPC seleccionados cumplen con los requerimientos de forma <u>EFICIENTE</u>	20
El documento de diseño esta <u>BIEN</u> escrito y tiene una estructura <u>CORRECTA</u>	20
La codificación del programa cumple con criterios de programación <u>DEFENSIVA</u> .	20
Para cada image el sistema muestra el tiempo preciso utilizado para ejecutar la convolución (no considere tiempos de lectura y escritura de archivo)	20

5. Checklist

Item	
El programa compila sin errores	
El programa es fácil de entender y tiene comentarios útiles	
El manejo de datos se hace con TDAs apropiadas según las operaciones a realizar sobre los datos.	

Los algoritmos utilizados minimizan el tiempo de procesamiento	
Las sentencias de repetición (while/for) emplean condiciones	
Las tareas del sistema se realizan en secciones independientes del programa	
Se minimiza el riesgo de inconsistencia de datos	
Los recursos se liberan cuando el programa termina	
Mi programa se ejecuta más rápido que una version secuencial que realice la misma tarea.	
Existe un mecanismo IHM para ingresar nuevas imagenes mientras otras se ejecuta	
No hay parámetros "quemados" en el programa	
Si un parámetro de entrada es incorrecto el programa no termina	