

Trabajo Práctico N° 1- Parte3

Comunicación y Concurrency

Fecha de entrega:

a. 08/05/2024

Forma de entrega:

- b. Se deberá generar un informe donde contenga los siguientes puntos
- **Carátula:** con los integrantes del grupo
 - **Link a un repositorio de github:** Generar un archivo de Colab por cada lenguaje utilizado y almacenarlos en este repositorio. Estos archivos deben ser con la extensión ipynb. También se debe subir a github el código fuente.
 - **Conclusiones:** En esta sección se debe describir las dificultades que encontraron al realizar el trabajo práctico
- c. Entregar el informe por plataforma MleL. Este debe ser en formato .pdf, con nombre TP1_Parte3_NumerodelGrupo.pdf.

Enunciado:

Realizar los siguientes ejercicios en el lenguaje de programación dado:

- **C++**

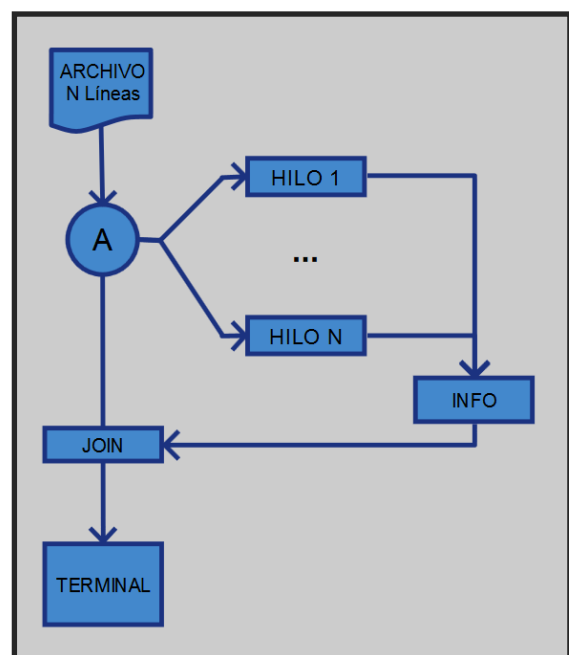
Crear un programa que lea un archivo de texto y genere la siguiente información:

- Cantidad de líneas efectivas (no vacías)
- Cantidad de palabras
- Cantidad de caracteres
- Cantidad de espacios
- Palabra con mayor ocurrencia y cantidad de veces que aparece

Para esto, el programa creará un hilo por cada línea no vacía y será tarea de los hilos generar la información requerida, cada hilo recolectará la información de su línea y la almacenará en un área común entre todos los hilos.

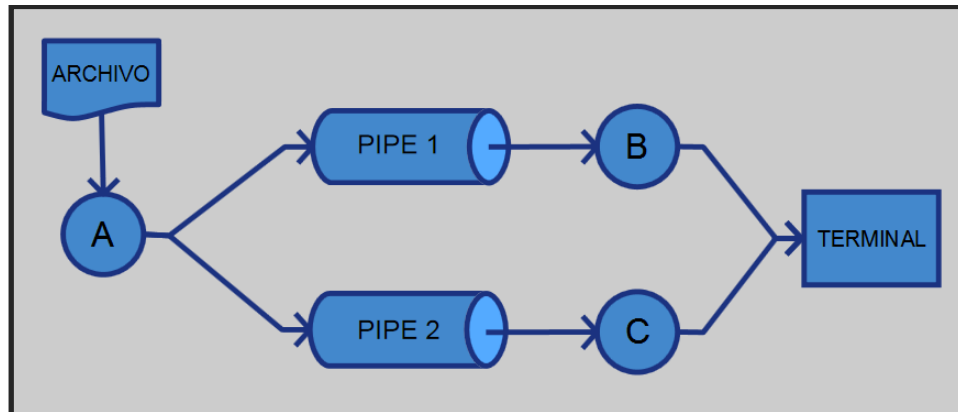
Por último, el hilo principal (main) informará por pantalla la información recopilada.

Nota: El programa recibirá por parámetro el path del archivo



- **Python**

Comunicar tres procesos emparentados A, B y C (Padre e Hijos) a través de tuberías (**Pipe**) como se detalla a continuación:



El programa A (Padre) leerá desde un archivo los registros de ventas de sus dos sucursales y enviará la información leída a los programas B y C (Hijos) a través de las tuberías. Los registros de la sucursal 1 serán enviados por el PIPE 1 hacia B y los de la sucursal 2 por el PIPE 2 hacia C.

El tipo del archivo (txt, csv, json, binario, etc.) con los registros de ventas, así como los productos comercializados queda a elección del alumnado y deberá almacenar la siguiente información por cada registro: SUCURSAL (1-2), FECHA (DD-MM-AAAA), PRODUCTO (Nombre), MONTO.

Por último, cada proceso hijo deberá procesar la información recibida generando la siguiente salida por pantalla al finalizar el envío de datos:

- Número de sucursal
- Monto total
- Producto más vendido
- Fecha con mayor cantidad de ventas y la cantidad (Máximo tres en caso de empate).
- Fecha con mayor monto vendido y el monto (Máximo tres en caso de empate).

Nota: Garantizar que informe por pantalla primero la sucursal 1, y luego la sucursal 2.

- **Java**

Problema del Zorro, el Pollo y el Maíz

Un granjero debe transportar un zorro, un pollo y una bolsa de maíz al otro lado de un río. El granjero tiene un bote pequeño y solo puede transportar un elemento a la vez. Sin embargo, si el zorro y el pollo se quedan solos, el zorro se come al pollo, de igual manera si el pollo queda solo con el maíz, se lo comerá.



¿Cómo resuelve el granjero este problema?

Crear un programa en java que utilizando concurrencia simule las situaciones posibles de dicho problema y su desenlace en base una entrada provista por el usuario. Por ejemplo, en el estado inicial se encuentran todos en el punto A:

> *Granjero, zorro, pollo y maíz se encuentran en el punto A*

El programa pregunta

> *¿Cruzar con?*

El usuario ingresa

> *Zorro*

El programa responde

> *Cruzando de punto A hacia B*

> *Granjero y zorro se encuentran en el punto B*

> *El pollo se come el maíz en el punto A*

Fin del programa

El modo de proveer al programa con las acciones, así como los mensajes informativos queda a criterio del alumnado, puede ser interactivo como en el ejemplo anterior, pasar la secuencia completa por parámetro, levantar desde un archivo, etc., lo importante es que el granjero pueda cruzar, acompañado o solo, desde el punto A hacia B y viceversa.

Nota: Queda a criterio del alumnado el diseño del programa, el único requisito es que se implementen hilos y algún mecanismo de sincronización entre ellos, si necesitaran implementar más de un mecanismo de sincronización así como mecanismos de comunicación, pueden hacerlo.