



CURSO: CMP 5002 - DATA MINING
COLEGIO: POLITÉCNICO
Semestre: 2do Semestre 2023/2024

Proyecto 1: Ejercicio usando la técnica **MapReduce**

Modalidad: Trabajo en equipo. Por tanto, cada integrante debe preparar su parte (asignada por el líder) y dominar la técnica evaluada completamente.

Problema:

1. Se desea implementar un programa que permita la lectura de un fichero (Word, texto) y devuelva el índice de palabras empleadas en el documento (la estructura del índice tiene que ser en el formato *<palabra, frecuencia>*).

Nota: La elección del documento es de libre albedrío con un peso mínimo de un 1gb. Se sugiere que sea en idioma inglés y que sea un texto coherente (libros, historias, dramas, novelas, etc). Para lograr el peso requerido del documento, se puede hacer varios *append* de la misma información al final del documento original y repetir este proceso varias veces hasta lograr la meta. Por ejemplo: documento final = documento final + documento original - ($\text{docF} = \text{docF} + \text{docO}$).

Requisitos:

- Es obligatorio el uso de la filosofía **MapReduce**, en este caso sobre una arquitectura de nodo simple (una PC con varios *core*), pero, paralelizable (usar hilos de programación) y distribuido (diferentes *pools* de hilos).
- Cargar al D2L la implementación del proyecto (fichero compactado) dentro del plazo de entrega.

Requisitos funcionales de la técnica:

- El documento debe ser dividido en ficheros (*chunks*) de hasta 32mb.
- La cantidad de nodos puede ser aleatoria (dependiendo de las prestaciones de sus computadoras), pero, con un mínimo de 2 nodos **reduce** y 4 nodos **map** (2 por cada **reduce**).
- Almacenar en ficheros texto, la salida de cada paso de la técnica **MapReduce**. Esto garantiza el NO uso de memoria interna.
- Garantizar una opción de fallo sobre los distintos nodos en tiempo de ejecución de la técnica **MapReduce**.
- Implementar un esquema exitoso de nodo **coordinator**, que sea capaz de asignar tareas y controlar el estado de los nodos **map** y **reduce**.

Por ejemplo: si mínimamente tienen 12 *chunks* del fichero de entrada y 4 nodos **map**, significa que cada nodo podrá procesar más de un *chunk* y esta labor debe ser controlada y coordinada por el nodo **coordinator**.

Evaluación:

- Fase 1:
 - Aplicar correctamente el **MapReduce** sobre un fichero de entrada y obtener la salida que da solución a la problemática planteada. (40% de la nota final de la tarea)
- Fase 2:
 - Aplicar elementos de paralelización y distribución correctamente en el **MapReduce** (30% de la nota final de la tarea)
- Fase 3:
 - Aplicar y corregir fallos a los distintos nodos: **coordinator**, **map**, y **reduce** (30% de la nota final de la tarea)
- Trazabilidad obligatoria:
 - En todas las fases se debe imprimir el resultado y para el caso de la fase 3, se debe, preguntar al usuario que nodo desean parar (inducir un fallo programáticamente)
- +1
 - Usar la técnica sobre un sistema distribuido (conectar mínimo dos PCs)

Nota: En cada fase de evaluación el profesor aplicará puntos de chequeo sobre el código implementado y basado en la trazabilidad. Se verificará el plagio en la implementación.