

Ejercicios

Ejercicio #1

Cree un programa que solicite al usuario una cantidad K de nodos de un dígrafo con pesos. Para cada uno de los nodos deberá crear entre 1 a 4 arcos a cualquier nodo seleccionado aleatoriamente sin repeticiones, a cada arco asígnele un peso aleatorio entre 4 a 14. Implemente dicho grafo usando listas de adyacencia. Proceda ahora a calcular los MST (Minimum Spanning Tree) de cada nodo del grafo. Nunca se usarán más de 200 nodos. Dicho cálculo deberá hacerlo por medio de un algoritmo en paralelo para que cada MST se calcule. Al final los MST deberán imprimirse en preorden en la consola mínimo.

Ejercicio #2

Existe una lista de canciones en un archivo CSV donde viene correctamente tabulado el nombre de la canción, el artista y el género. Existe un folder con 100 archivos mp3 que algunos tienen problemas en el nombre de la canción, el nombre del archivo, el género y/o el artista. Cree un algoritmo paralelo que logre arreglar la información tag de los mp3 usando como base de comparación la base de datos proporcionada por los CSV. Deberá tratar de hacer matching de la cantidad de información posible, manejar aproximaciones de palabras, palabras incompletas y control de lowercase.

Ejercicio #3

Cree un programa que se conecte a Facebook con su usuario y en paralelo cree un reporte que muestre los amigos a los que no les ha hablado durante la semana en curso, durante el mes en curso o durante el año en curso.

Implementación

El ejercicio número #1 deberá hacerse con afinidad de procesador en java.

El ejercicio número #2 se debe hacer usando el fork/join framework de java
<http://docs.oracle.com/javase/tutorial/collections/streams/parallelism.html>

El ejercicio número #3 se debe hacer con afinidad de procesador en .Net

En todos los ejercicios debe garantizarse que se usen todos los procesadores disponibles que le computador posea.

El trabajo puede hacerse en parejas. Revisión contra cita de entrega el lunes 23 de Junio.