

1 Descripción del problema

Uno de los algoritmos más importantes para lograr la compresión de archivos es el de árboles de Huffman. Para ello se puede ver el detalle del algoritmo en este enlace:

https://es.wikipedia.org/wiki/Algoritmo_de_Huffman.

El problema consiste en poder programar ese algoritmo paralelizando aquellas fases que pueden ser paralelizadas.

1.1 Fase 1: Conteo de frecuencias

Esta parte debe ser totalmente paralelizada y con la cantidad de hilos pasada por parámetro al iniciar el programa. Si se reciben n hilos, entonces debe de separar el archivo en n partes tratando de que cada una de las partes tenga la misma cantidad de bytes, de modo que las cargas de trabajo se repartan equitativamente.

Cada uno de los hilos debe de contar las frecuencias de cada byte y generar una única tabla de frecuencias final (con la suma de todos los hilos, ésta debe ser la única parte que requiera de sincronización y manejo de hilos).

1.2 Fase 2: Creación del árbol

Para esta parte, se tendrá como insumo la tabla de frecuencias general donde se generarán los nodos hijos de cada dos nodos menos frecuentemente representados.

1.3 Fase 3: Generación de los códigos

Esta parte debe de ser parcialmente paralelizada, de modo que 4 hilos independientes puedan encontrar los códigos de las hojas. Cada hilo debe de buscar en una rama diferente del árbol, para ello se separarán por los hilos 00, 01, 10 y 11. Si se diera que una de las ramas no existiera (es decir, que hay algún carácter que se repite en al menos $\frac{1}{4}$ del archivo, simplemente esos hilos no se generan.

1.4 Fase 4: Compresión

La parte de compresión debe ser también paralelizada, una vez que se cuenta con la tabla de códigos de huffman, entonces se debe de generar

los bits de compresión, que consiste en la concatenación binaria de cada una de las partes de los códigos de huffman en el mismo orden que se generaron en las diferentes partes del archivo y se escribe el archivo en **binario**. El nuevo archivo debe adjuntar el nombre *.edy* al final del archivo original. Así si el nombre del archivo era *Archivo.txt* el compreso debería decir: *Archivo.txt.edy*. Además deberá de guardar la tabla de códigos de huffman en un archivo de nombre *Archivo.txt.huff*

Finalmente se compactan todas las compresiones y se genera el archivo final.

2 Entradas del programa

El progrma recibe de consola 2 parámetros, el nombre del archivo y el número de hilos

3 Salidas del programa

El programa debe de mostrar en pantalla algunos datos durante su ejecución:

Tabla de frecuencias completa Cantidad de hojas del árbol Tiempo de ejecución de cada etapa.

4 Descompresor

Finalmente y para validad que todo está correctamente hecho, se procederá a descomprimir los archivos, este proceso no se hace en paralelo y bastará con que se pueda generar el archivo original con base en los dos que se guardaron.

5 Evaluación

La evaluación queda a criterio del profesor en cuanto a la completitud y funcionalida de cada una de las fases.