Algoritmia Básica

Práctica 1: Compresor de ficheros basado en el código Huffman

Andrés Gavín Murillo 716358
Andrew Mackay 737069

Descripción del problema

Se trata de un problema de compresión y descompresión de ficheros mediante el algoritmo de Huffman. Para ello se utiliza un código de longitud variable, dando codificaciones cortas a los símbolos más frecuentes y largas a los menos frecuentes.

Decisiones tomadas

Al comienzo del fichero de salida se almacena el árbol lexicográfico de código libre. Primero almacena todas las frecuencias y luego todos los símbolos, recorriendo el árbol en in-orden.

Los símbolos son de 1 byte cada uno.
Los símbolos codificados equivalen a un número de bits (menores, mayores o iguales a 1 byte, dependiendo del símbolo codificado). Los bits resultantes se almacenan en un buffer hasta poder separarlos en bytes y volcarlos al fichero de salida. Como pueden quedar los últimos bits sin que lleguen a formar 1 byte, al final del fichero de salida se guarda el número de bits descolocados (para su posterior decodificación).

Se ha creado un TAD de tipo árbol, que representa un árbol binario, formado por nodos (valor: símbolo, clave: frecuencia) que a su vez son del tipo árbol para almacenar el árbol lexicográfico.

Se ha creado un heap de mínimos para crear el árbol lexicográfico, es decir, se obtienen los nodos hoja correspondientes a cada símbolo y se insertan en el heap. A continuación, se extraen del heap los dos nodos con menor frecuencia y se unen (nodo padre que suma sus frecuencias), insertando el nodo resultante al heap. Finaliza cuando en el heap solo queda un nodo, que es el árbol lexicográfico de código libre de prefijos óptimo.

Pruebas realizadas y análisis de resultados

Para comprobar el funcionamiento del algoritmo y analizar su eficacia, se han realizado varias pruebas con ficheros de distinto tamaño. Concretamente, se han utilizado ficheros con tamaños de 10KB a 1MB descargados de la página https://sample-videos.com/download-sample-text-file.php. Por otro lado, también se ha utilizado un fichero que contiene una versión de “El Quijote” de 2.19MB obtenido en: https://www.gutenberg.org/cache/epub/2000/pg2000.txt. También se ha comprimido el ejecutable “huf” para comprobar que funcionaba correctamente una vez descomprimido. Las pruebas han sido realizadas en Hendrix y los tiempos se han obtenido realizando una media de 10 intentos (mediante la herramienta “time”, y utilizando el resultado “real”). Los resultados son los siguientes:

Fichero	Tamaño (KB)	Tamaño Comprimido (KB)	Tiempo en comprimir (segundos)	Tiempo en descomprimir (segundos)
10kb.txt	9.3	5.5	0.057	0.048
20kb.txt	19	11	0.064	0.073
50kb.txt	49	27	0.119	0.100
100kb.txt	100	54	0.189	0.164
200kb.txt	199	107	0.334	0.232
500kb.txt	499	268	0.724	0.467
1000kb.txt	999	537	1.347	0.848
quijote.txt	2100	1200	2.860	1.781
huf	33	27	0.132	0.087