Práctica Número 3: Compresión de ficheros de texto mediante códigos de Huffman

1. Introducción

El algoritmo de Huffman es un algoritmo para la construcción de códigos de Huffman, desarrollado por David A. Huffman en 1952 y descrito en *A Method for the Construction of Minimum-Redundancy Codes*[2].

Existen multitud de referencias a los árboles de Huffman en la literatura, entre otros en [1], así como en Internet, por ejemplo en la wikipedia o por ejemplo en este blog. Por esa razón, omitimos explicaciones teóricas y vamos trabajar directamente un ejemplo práctico.

Todos sabemos que el codigo ascii, representa cada caracter con 8 bits (ascii extendido). Ascii trata todos los caracteres como iguales, por eso los codifica todos con 8 bits, pero por lo general, hay caracteres que se repiten más que otros. Como podéis ver en este enlace de wikipedia, en Castellano lo que más se repite es el espacio y, en orden descendiente, las letras e, a, o, s, r, n ... En esta práctica vamos a conseguir, utilizando árboles, asignar un número de bits menor a los caracteres que más se repiten; de esa forma vamos a conseguir que el texto ocupe menos espacio, para así comprimirlo y ahorrar espacio en disco.

2. Ejemplo y tareas a realizar

Tenemos el siguiente texto "En un lugar de...". El alfabeto del texto, es decir, las letras que lo componen son: "En ulgarde." incluyendo el espacio y el punto.

Una vez tenemos las letras del alfabeto, contamos su frecuencia de aparición, que de forma ordenada de menor a mayor es:

E	l	g	a	r	d	e	n	u		.
1	1	1	1	1	1	1	2	2	3	3

El siguiente paso, es crear un vector de árboles. En cada árbol, tendremos de momento una sola letra como dato y como clave de búsqueda su frecuencia de aparición.

Consultando las notas de la asignatura y siguiendo el método allí descrito, se pide construir manualmente los códigos de Huffman asociados a los caracteres y frecuencias anteriores y dar el texto comprimido.

Para ello se siguen los pasos siguientes:

1. Obtener las frecuencias de las letras.
2. Crear el vector de árboles con las letras y las frecuencias.
3. Ordenar el vector de árboles inicial.

4. Crear el árbol de Huffman.
5. Obtener el diccionario.
6. Comprimir.

A continuación debes descargarte el material suministrado y se pide

1. Diseñar e implementar un experimento que permita inferir la proporción de compresión media en strings generados aleatoriamente
2. Diseñar e implementar un experimento que permita inferir la proporción de compresión media en textos en distintos rangos.

Referencias

[1] T. H. Cormen, C. E. Leiserson, R.L. Rivest, C. Stein, *Introduction to Algorithms.* The MIT Press, second edn., 2001.

[2] D.A. Huffman, "A Method for the Construction of Minimum-Redundancy Codes", *Proceedings of the I.R.E.*, September 1952, 1098–1102