



Taller 5: Estructuras de datos

Fundamentos de análisis y diseño de algoritmos

Carlos Andres Delgado S, Msc
cadelgado1@usbcali.edu.co

Noviembre de 2021

Reglas del taller

1. El taller pueden ser realizado por grupos de hasta 4 personas
2. Entregue la solución de los puntos del taller en un archivo comprimido con los archivos de Python que implementó y un informe en formato PDF. Incluya los códigos desarrollados en el archivo.
3. Entregue el taller en el enlace dispuesto por el campus virtual.
4. Se debe hacer una entrega por grupo, en caso de tener varias entregas por grupo se valdrá únicamente la primera que el docente revise
5. El plazo para entregar el taller será Domingo 21 de Noviembre de 2021 a las 23:59:59 hora de Colombia, si entrega después de esta fecha recibirá una penalización

1. Codificación Huffman

Existen muchos métodos para la compresión de datos, uno de estos es la codificación Huffman. Este método consiste en generar una tabla para codificar un determinado símbolo. Este método fue desarrollado por David A. Huffman mientras era estudiante y publicado en *A Method for the Construction of Minimum-Redundancy Codes*[1]. El algoritmo de Huffman utiliza una distribución de probabilidad uniforme para codificar los símbolos.

En este taller usted deberá implementar el algoritmo para codificar archivos de texto que contienen los siguientes símbolos:

$$\{a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y, z, \\ 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, \text{punto}, \text{espacio}, \text{coma}\}$$

No se diferencian mayúsculas de minúsculas. El algoritmo puede agrupar los símbolos para mejorar la compresión, sin embargo para efectos de este taller sólo considere un símbolo a la vez.

1.1. Trabajo a realizar

1.1.1. Codificación Huffman

1. Investigue cómo realizar la codificación de Huffman. Puede ayudarse con la bibliografía donde hay algunas referencias relevantes. Explique formalmente el algoritmo de codificación.
2. Muestre un ejemplo de la codificación.
3. Implemente la codificación:
 - Explique la estrategia de búsqueda símbolos que usted implementó.
 - Que estructuras de datos usted utiliza en su implementación. ¿Porque la eligió?
 - Analice las complejidades espacial y temporal, del algoritmo implementado y de las estructuras de datos utilizada. ¿Existe un mejor y peor caso?

Para la implementación en este punto de los árboles de etiquetas, no utilice árboles AVL:

1.1.2. Decodificación

1. Investigue cómo realizar la decodificación de un texto codificado por Huffman al original.
2. Muestre un ejemplo de la decodificación
3. Implemente la decodificación:
 - Que estructuras de datos usted utiliza en su implementación. ¿Porque la eligió?
 - Analice las complejidades espacial y temporal, del algoritmo implementado y de las estructuras de datos utilizada. ¿Existe un mejor y peor caso?

1.2. Formato de entrada

Es un archivo de texto, al cual se le debe aplicar la codificación o decodificación. En la aplicación debe solicitarse el nombre del archivo que contiene el texto codificado y el árbol de etiquetas asociado.

1.3. Formato de salida

Para el caso de la codificación la salida es:

1. Un archivo con la codificación binaria generada.
2. Un archivo con la codificación ASCII generada, recuerde que cada código binario se puede codificar como un carácter ASCII.
3. Un archivo con la siguiente información:

```
Nodos creados: X
Profundidad máxima del árbol generado: Y
Tabla de codificación generada:
      Simbolo ---- Codigo
S1  ----- C1
.
.
.
SN          CN
```

Para el caso de la decodificación la salida es un archivo de texto que contenga la información decodificada.

Referencias

- [1] D. Huffman, “A method for the construction of minimum-redundancy codes,” *MIT*, 1952.
- [2] Wikipedia, “Huffman Coding Article.” http://en.wikipedia.org/wiki/Huffman_coding, 2021.
- [3] Wolfram, “Huffman Coding Mathematica.” <http://mathworld.wolfram.com/HuffmanCoding.html>, 2021.
- [4] M. David, “Huffman Coding.” <https://www.cs.cf.ac.uk/Dave/Multimedia/node210.html>, 2021.
- [5] J. Morris, “Huffman Encoding.” <https://www.cs.auckland.ac.nz/software/AlgAnim/huffman.html>, 2021.