Estructures de dades
Curs 2020-21
Práctica 2 - Códigos de Huffman

8 d'abril de 2021

Fecha de entrega: 16/05/2021; 23:59h

Esta práctica consiste en implementar la **codificación Huffman** y trabajaremos en ella durante 4 semanas (podéis consultar el cronograma en la sección 5).

La realización de esta práctica puede ser **individual o en parejas**. La entrega constará de una carpeta comprimida que contendrá el proyecto GPS con el código fuente (en caso de no disponer de proyecto GPS, debe entregarse una carpeta comprimida con los ficheros .ads y .adb).

1 Introducción codificación Huffman

La codificación Huffman es un algoritmo utilizado para la compresión de datos. El término se refiere al uso de una tabla de códigos de longitud variable para codificar un determinado símbolo (como puede ser un carácter en un archivo), donde la tabla ha sido rellenada basándose en la probabilidad estimada de aparición de cada posible valor de este símbolo.

La codificación Huffman utiliza un método específico para elegir la representación de cada símbolo que da lugar a un código prefijo (es decir, la cadena de bits que representa un símbolo en particular nunca es prefijo de la cadena de bits de un símbolo diferente), el cual representa los caracteres más comunes utilizando las cadenas de bits más cortas.

En la Tabla 1 se puede consultar un ejemplo de tabla de frecuencias y su codificación asociada.

Carácter	Frecuencia	Código
a	5	1100
b	9	1101
c	12	100
d	13	101
e	16	111
f	45	0

Taula 1: Ejemplo de tabla de frecuencias y codificación asociada.

2 Algoritmo general

La codificación Huffman consiste en la creación de un mapping implementado con un árbol binario donde las etiquetas son caracteres y los valores son sus frecuencias (**árbol de Huffman**).

Los pasos para calcular el árbol de Huffman son:

1. Crear una tabla de frecuencias de un texto.
2. Crear tantos árboles de un solo nodo como caracteres aparecen en el texto. La clave será el carácter y el valor será la frecuencia.
3. Insertar todos los árboles en un Heap.
4. Mientras queden dos o más elementos en el heap:
   (a) Extraer los dos árboles con menos frecuencia.
   (b) Crear el árbol unión: el valor de la nueva raíz será la suma de las frecuencias de las raíces de los árboles seleccionados con menos probabilidad y el árbol que tenga la raíz con menos frecuencia corresponderá al hijo izquierda.
   (c) Insertar el nuevo árbol en el heap.

En la Figura 1 puede consultarse el árbol de Huffman construido a partir de la tabla de frecuencias de la Tabla 1.

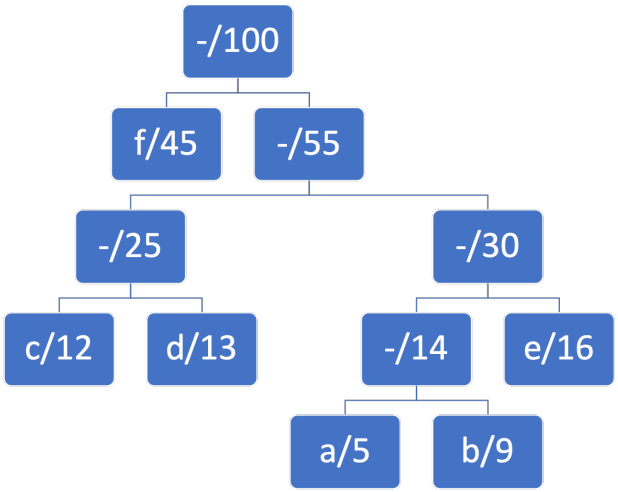

Figura 1: Árbol de Huffman asociado a la tabla de frecuencias de la Tabla 1.

Una vez se ha construido el árbol, la construcción del código se basa en etiquetar las aristas que unen cada uno de los nodos con ceros y unos, hijo izquierdo y derecho, respectivamente (ver Figura 2). El código resultante para cada carácter es la lectura, siguiendo la rama, desde la raíz hasta el nodo hoja correspondiente a cada carácter (ver Tabla 2).

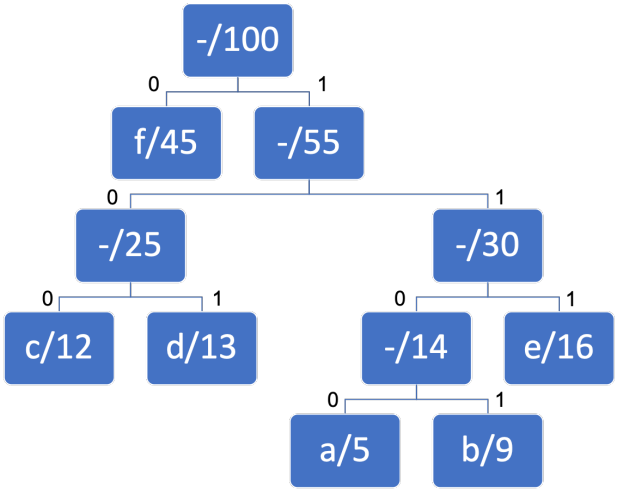

Figura 2: Árbol de Huffman con las aristas etiquetadas.

En la dirección `https://www.youtube.com/watch?v=0kNXhFIEd_w` podéis encontrar mas información respecto a la codificación Huffman.

Carácter	Código
a	1100
b	1101
c	100
d	101
e	111
f	0

Taula 2: Codificación asociada al árbol de Huffman de la Figura 2.

3 Estructuras necesarias

- Mapping implementado con un array (tabla de frecuencias).
- Árbol binario.
- Heap.

4 Propuesta de práctica

Construir un programa que tenga como entrada un archivo de texto llamado *entrada.txt*, construya el árbol de Huffman correspondiente al texto del archivo de entrada y genere como resultado un archivo de texto llamado *entrada_codi.txt* que contenga todos los caracteres que aparecen en el texto (en orden de abecedario) y su código binario correspondiente (un carácter y su código asociado por línea).

Por ejemplo, si el contenido de vuestro archivo *entrada.txt* es:

```
aixo es un exemple darbre de huffman
```

Una vez finalizada la ejecución, el archivo *entrada_codi.txt* debería contener:

```
 : 110
a: 1011
b: 10000
d: 1001
e: 111
f: 0011
h: 10101
i: 10001
l: 10100
m: 0100
n: 0101
```

```
o: 00010
p: 00011
r: 0010
s: 0000
u: 0111
x: 0110
```

5 Cronograma

Las tareas a implementar se encuentran distribuidas durante cuatro semanas:

- Semana 12/04/2021-18/04/2021: Construcción de la tabla de frecuencias a partir del texto contenido en el archivo *entrada.txt*.
- Semana 19/04/2021-25/04/2021: Creación de la cola de prioridad a partir de los árboles binarios de un solo nodo.
- Semana 26/04/2021-02/05/2021: Creación del árbol Huffman.
- Semana 03/05/2021-09/05/2021: Codificación a partir del árbol de Huffman.