

**LABORATORIO 7 – GUIA 1
CURSO ESTRUCTURAS DISCRETAS II**

Apellidos y Nombres:
 CUI: email (Institucional):
 Fecha: 26 de octubre 2022 Sección: A y B
 Docente: Dra. Roxana Flores Quispe /Dr. Yuber Elmer Velazco Paredes

Guía N° 1 : Arboles de Huffman

Alumno:

Nota:

I. Objetivos

- Presentar al alumno la implementación de arboles binarios de Huffman en C++.

II. Equipos y materiales

Equipos y dispositivos:

- PC.

Software:

- C++.

III. Actividades

- a) Crear un archivo de texto plano formado por la combinación de letras "a", "b", "c", "d", "e" y "f" en las siguientes cantidades:

Letras	a	b	c	d	e	F
Cantidades	45	13	12	16	9	5

- b) Crear una estructura de datos como se muestra a continuación:

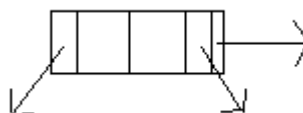


Figura 1. Nodo

```
struct nodo{
    char letra;
    int cantidad;
```

```
struct nodo *sig;
struct nodo *izq, *der;
};
typedef struct nodo *Inicio;
```

- c) Leer el archivo de texto plano creado en el paso 1 y crear una lista ordenada. Observe que la lista se encuentra ordenada por la frecuencia de aparición de los caracteres.

Se sugiere seguir los siguientes pasos:

- Conforme se vaya leyendo el archivo, crear el nodo y almacenar las frecuencias de aparición.
- Aplicar un algoritmo de ordenación para obtener la lista ordenada en base a la frecuencia.

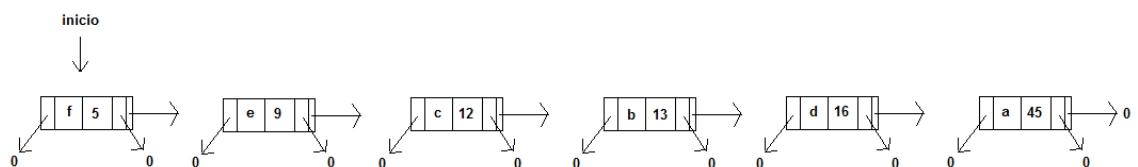


Figura 2.

- d) Diseñe un algoritmo de tal manera que forme un árbol de Huffman.

Se sugiere:

- Crear un nuevo nodo.
- Leer los dos primeros elementos de la lista
- Formar el árbol
- Ajustar los punteros, y
- Reordenarlo en la lista.

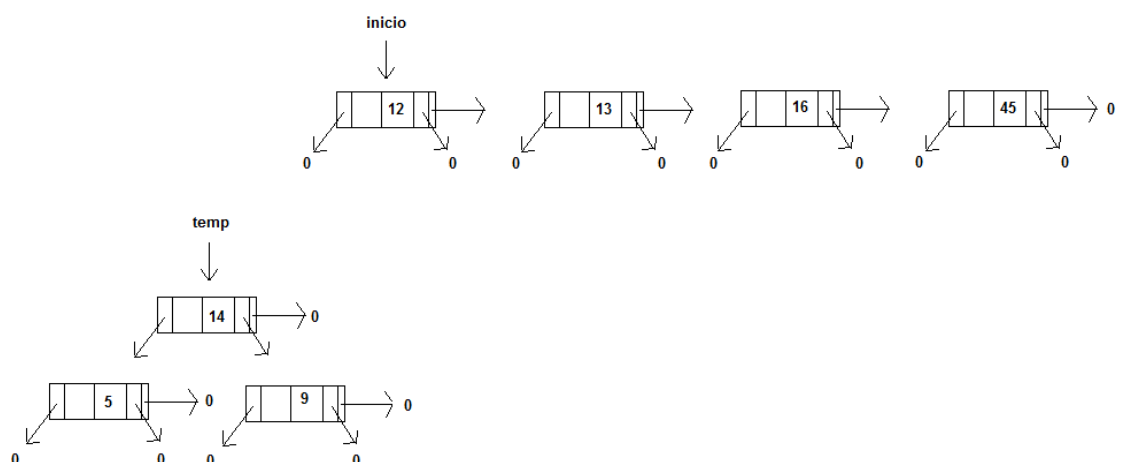


Figura 3.

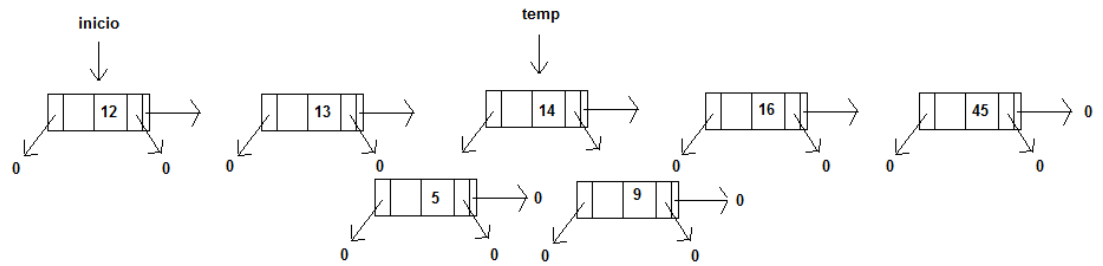


Figura 4.

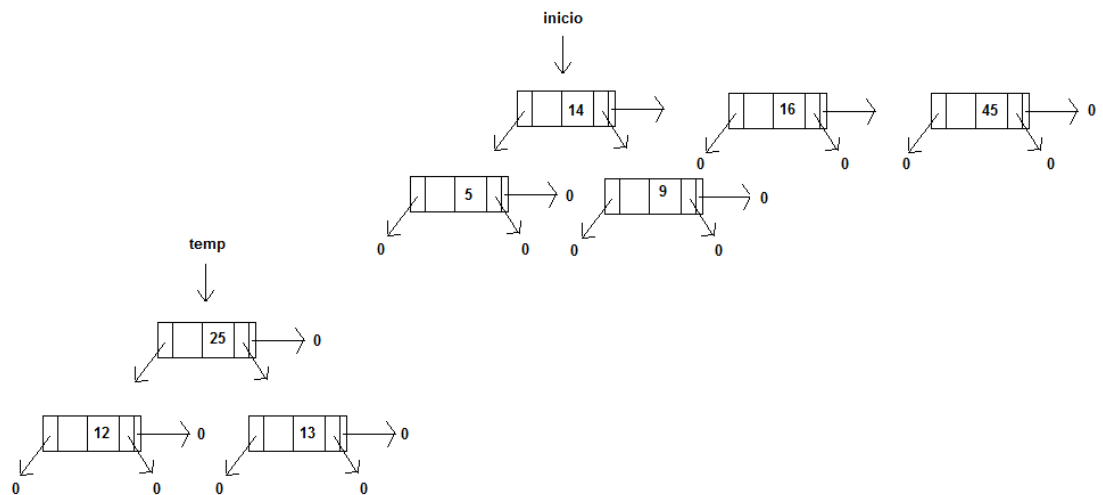


Figura 5.

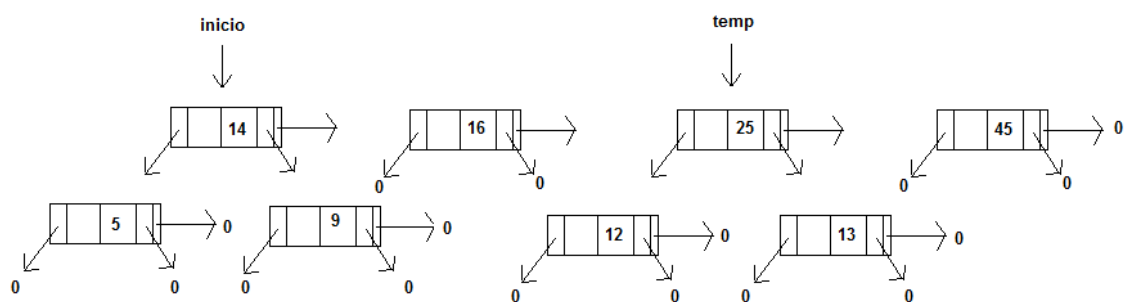


Figura 6.

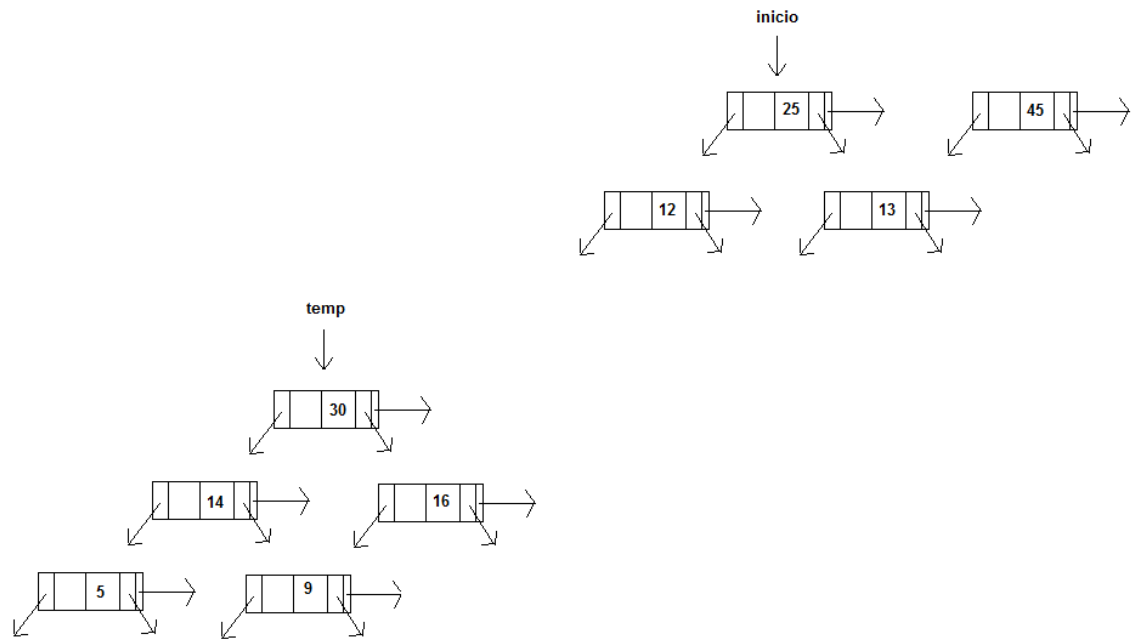


Figura 7.

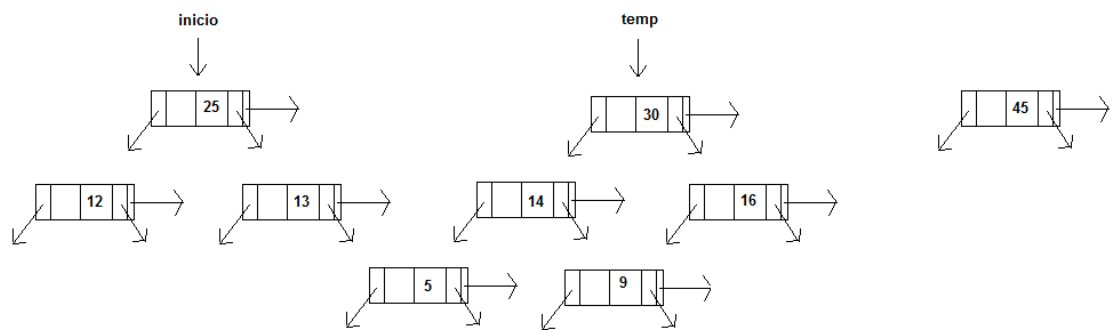


Figura 8.

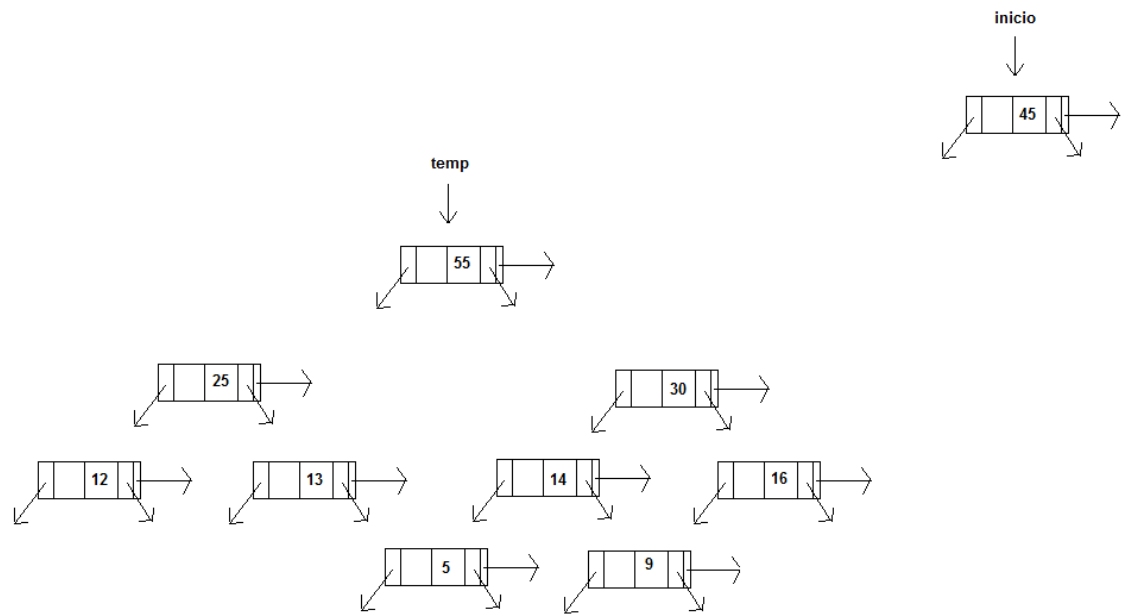


Figura 9.

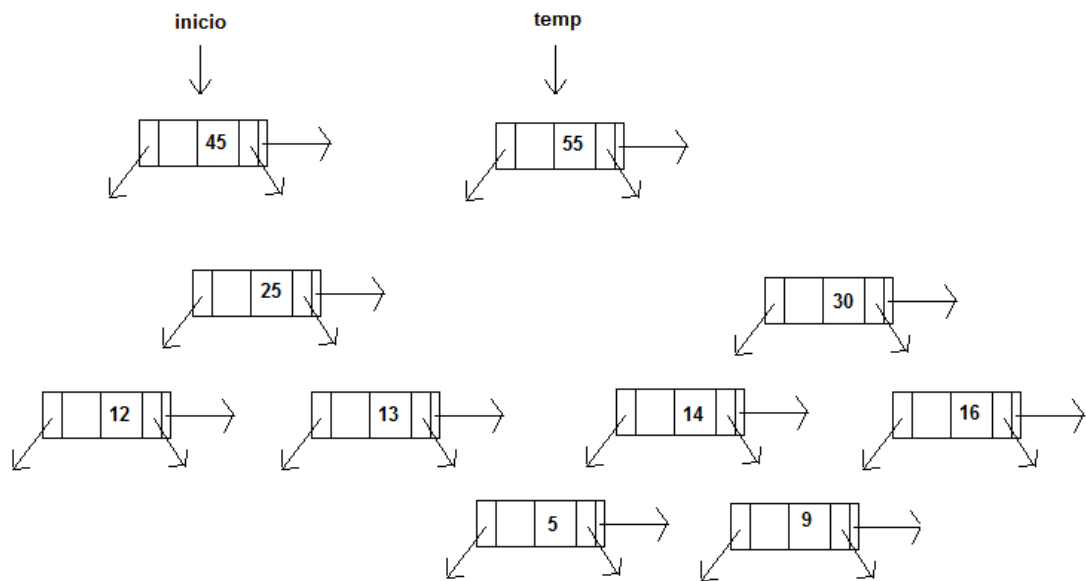


Figura 10.

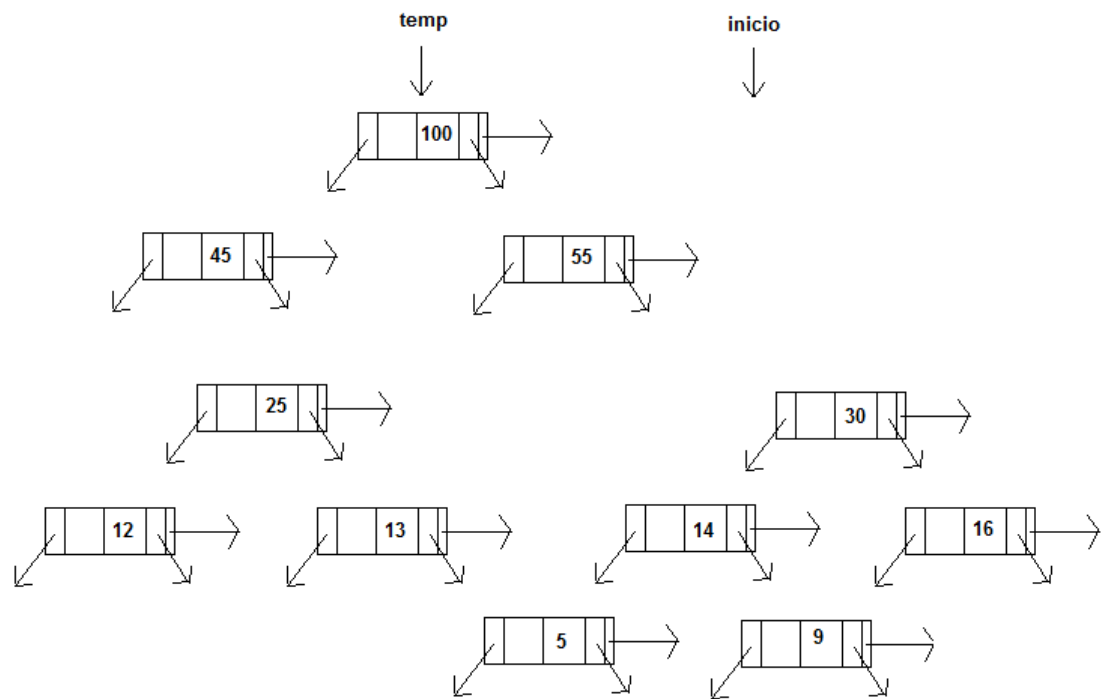


Figura 11.

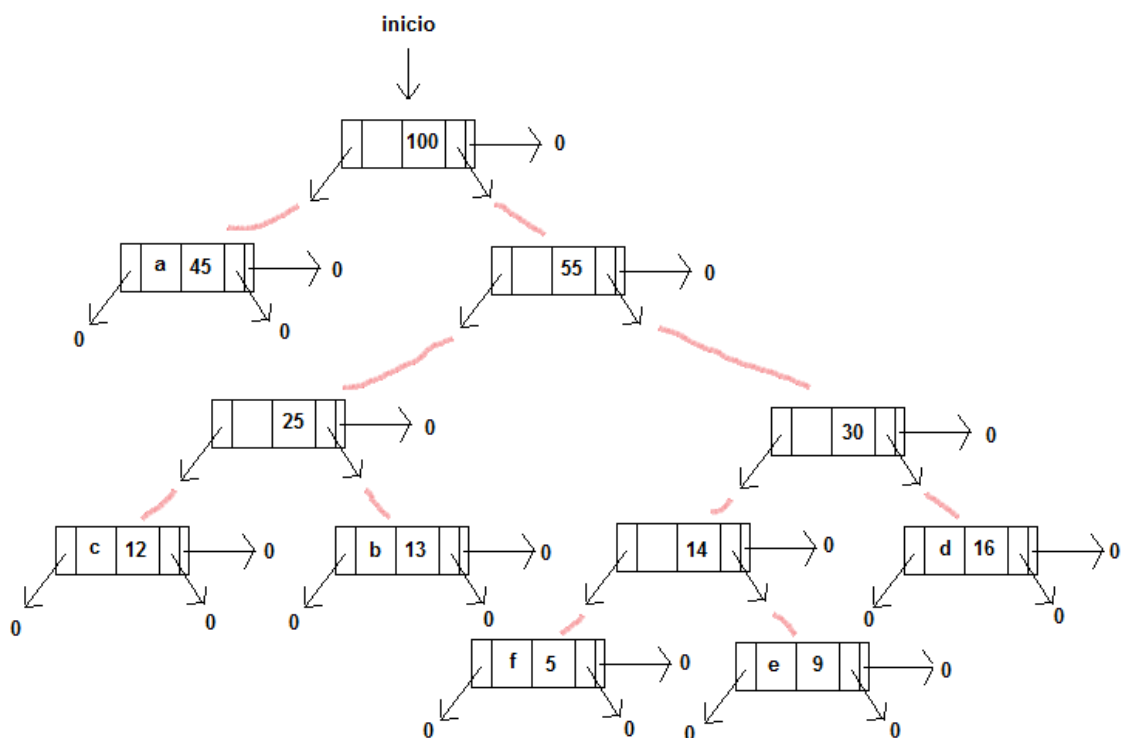


Figura 12.

- e) Recorra el árbol y reconozca las siguientes cadenas.
W1 = 000
W2 = 100011001101
W3 = 110001001101
- f) Proponga y diseñe un algoritmo, en referencia al árbol de Huffman, que codifique un archivo de texto plano.
- g) Proponga y diseñe un algoritmo, en referencia al árbol de Huffman, que decodifique un archivo de texto plano.
- h) Proponga y diseñe un algoritmo, que permita grabar y recuperar el árbol de Huffman en un archivo de texto plano.

IV. Resultados

El alumno comprende la importancia de los árboles de Huffman, tanto para compresión como para encriptación de datos.