

# **Guía de usuario**

## **Técnicas voraces**

**Alumnos:**

**De la Mora Villaseñor Diego Gabriel**

**Lopez Esparza Angel Emanuel**

**Lopez Galván Melanie Montserrat**

**Código: 223991772**

**Materia: Análisis de algoritmos**

**Sección: D06**

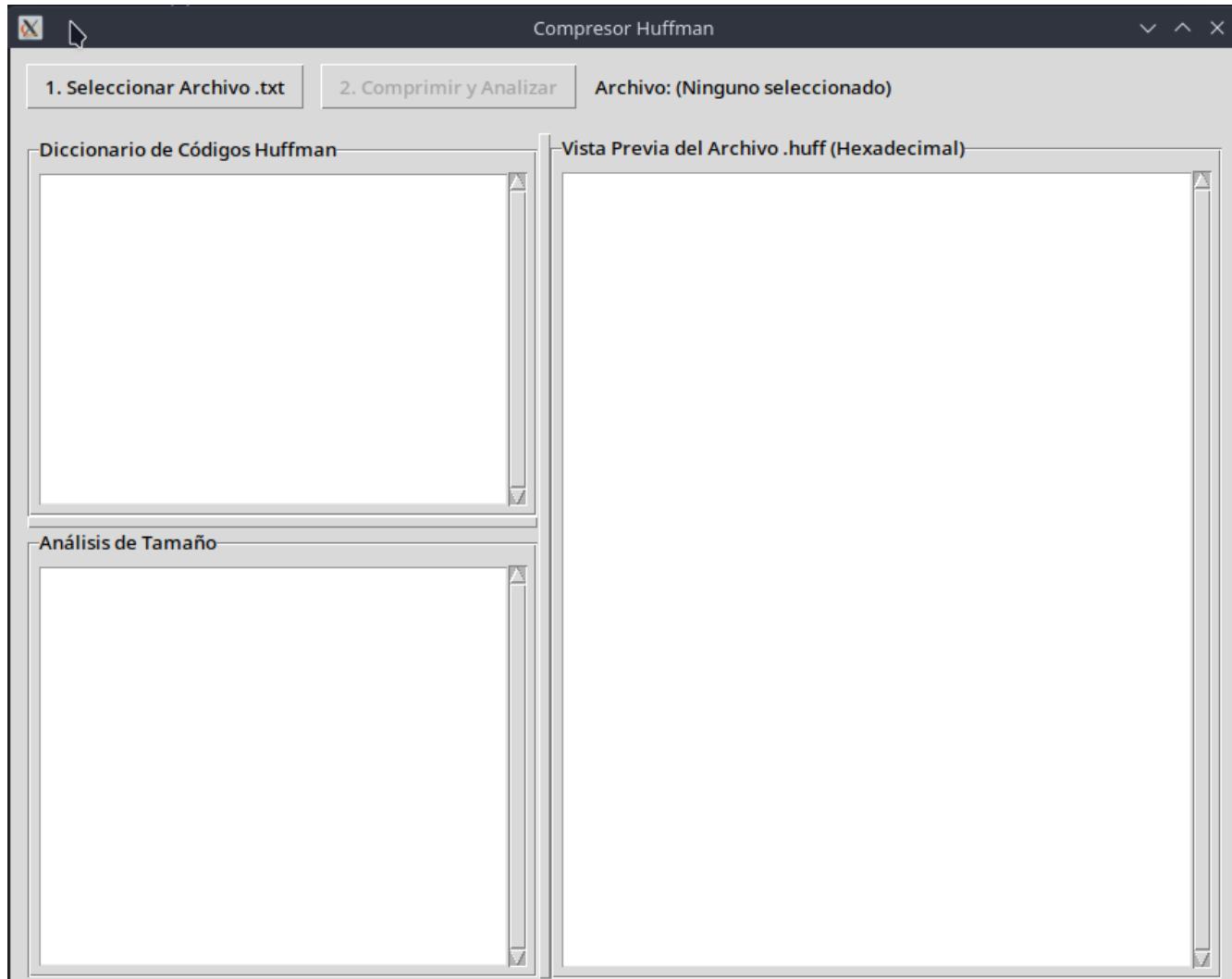
**Profesor: Lopezarce Delgado Jorge Ernesto**



# Guía de usuario

Esta herramienta utiliza la codificación de Huffman (una técnica voraz) para comprimir archivos de texto .txt. El programa toma un archivo de texto, analiza la frecuencia de sus caracteres, y genera un archivo binario comprimido con la extensión .huff.

Inmediatamente después, el programa realiza el proceso inverso: lee el archivo .huff, lo decodifica y crea un nuevo archivo .txt que debe ser idéntico al original, demostrando una compresión sin pérdida de datos.



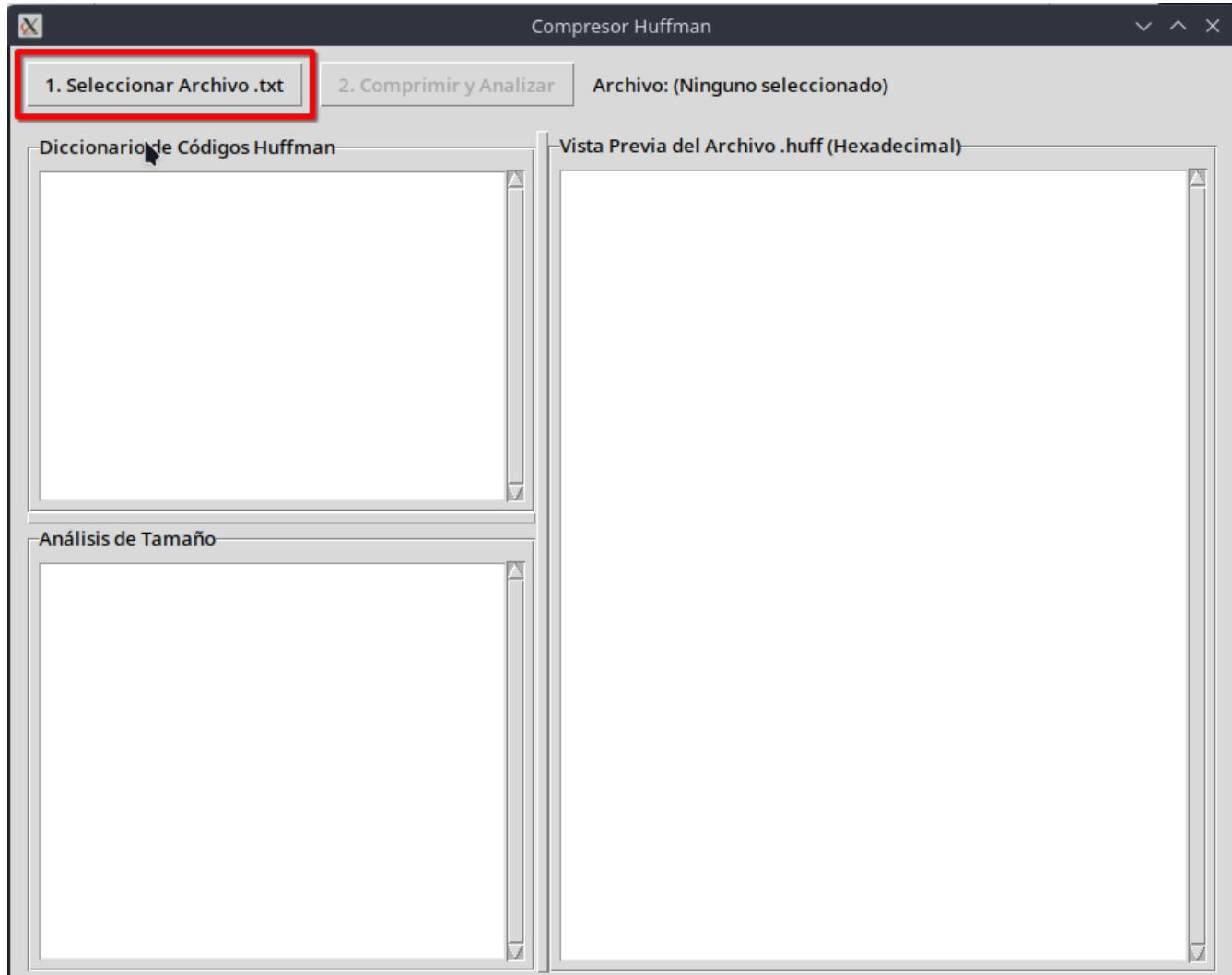
## 1. El Proceso Completo: Comprimir y Verificar

Tu aplicación realiza todo el ciclo (comprimir, descomprimir y verificar) en un solo proceso para demostrar la efectividad del algoritmo.

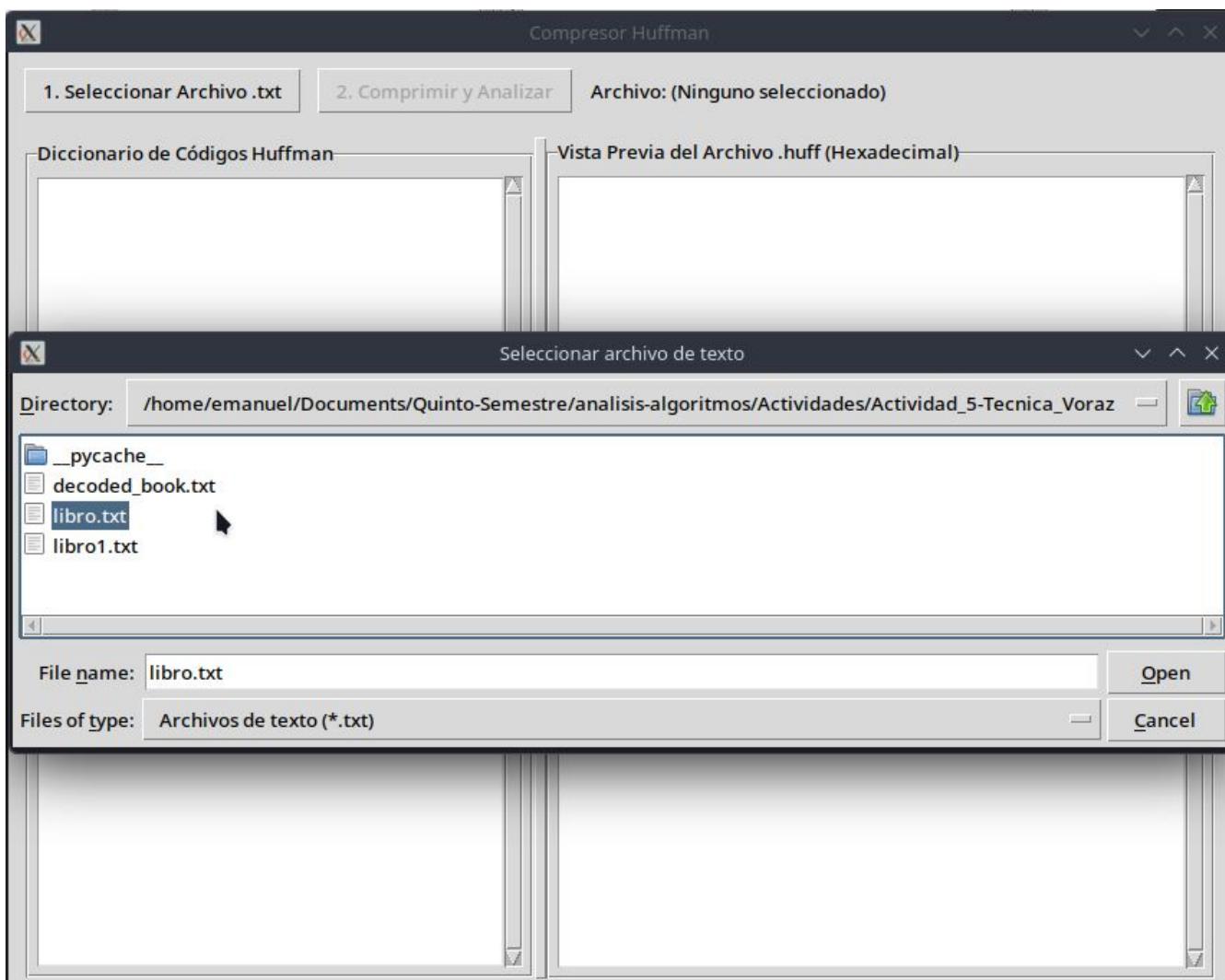
### Paso 1: Seleccionar el Archivo de Texto Original

Para comenzar, necesitas indicarle al programa qué archivo .txt deseas comprimir.

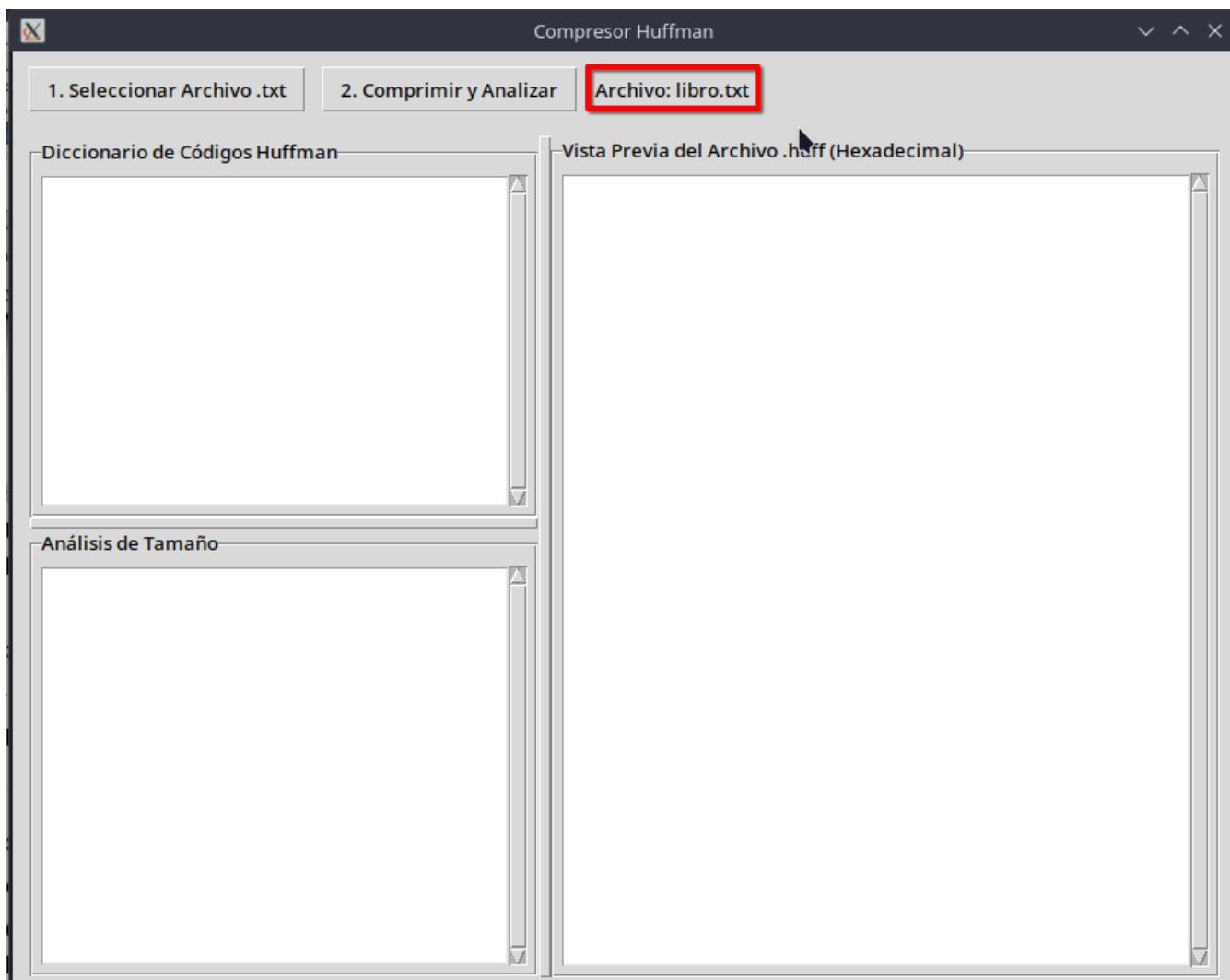
1. Haz clic en el botón "Seleccionar Archivo.txt".



2. Se abrirá un explorador de archivos. Navega hasta tu archivo de texto (por ejemplo, libro.txt) y selecciónalo.



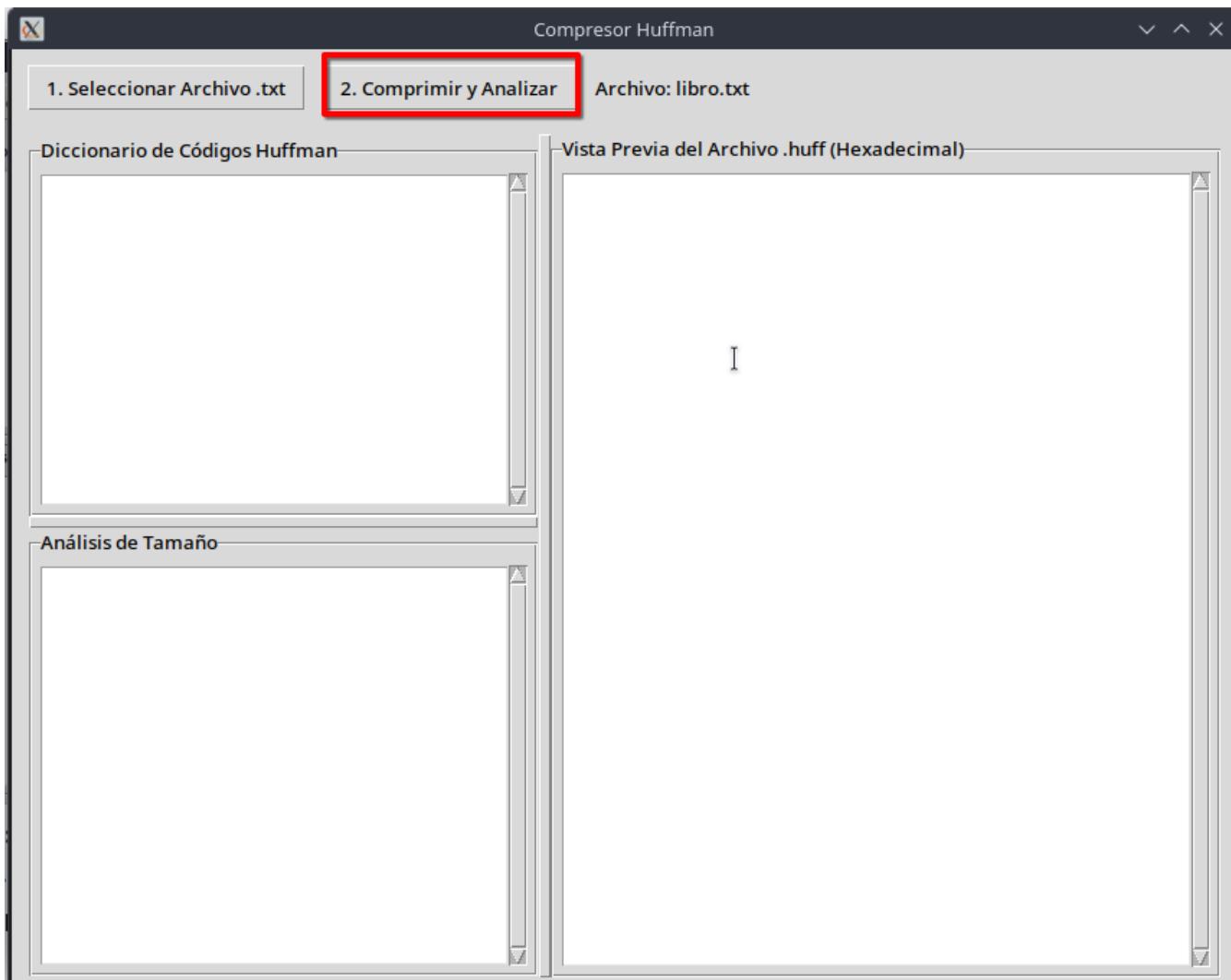
3. Una vez seleccionado, verás el nombre del archivo en la parte superior derecha:



## Paso 2: Iniciar el Proceso

Con el archivo cargado, estás listo para ejecutar el algoritmo.

1. Haz clic en el botón principal de acción, "Comprimir y Analizar".



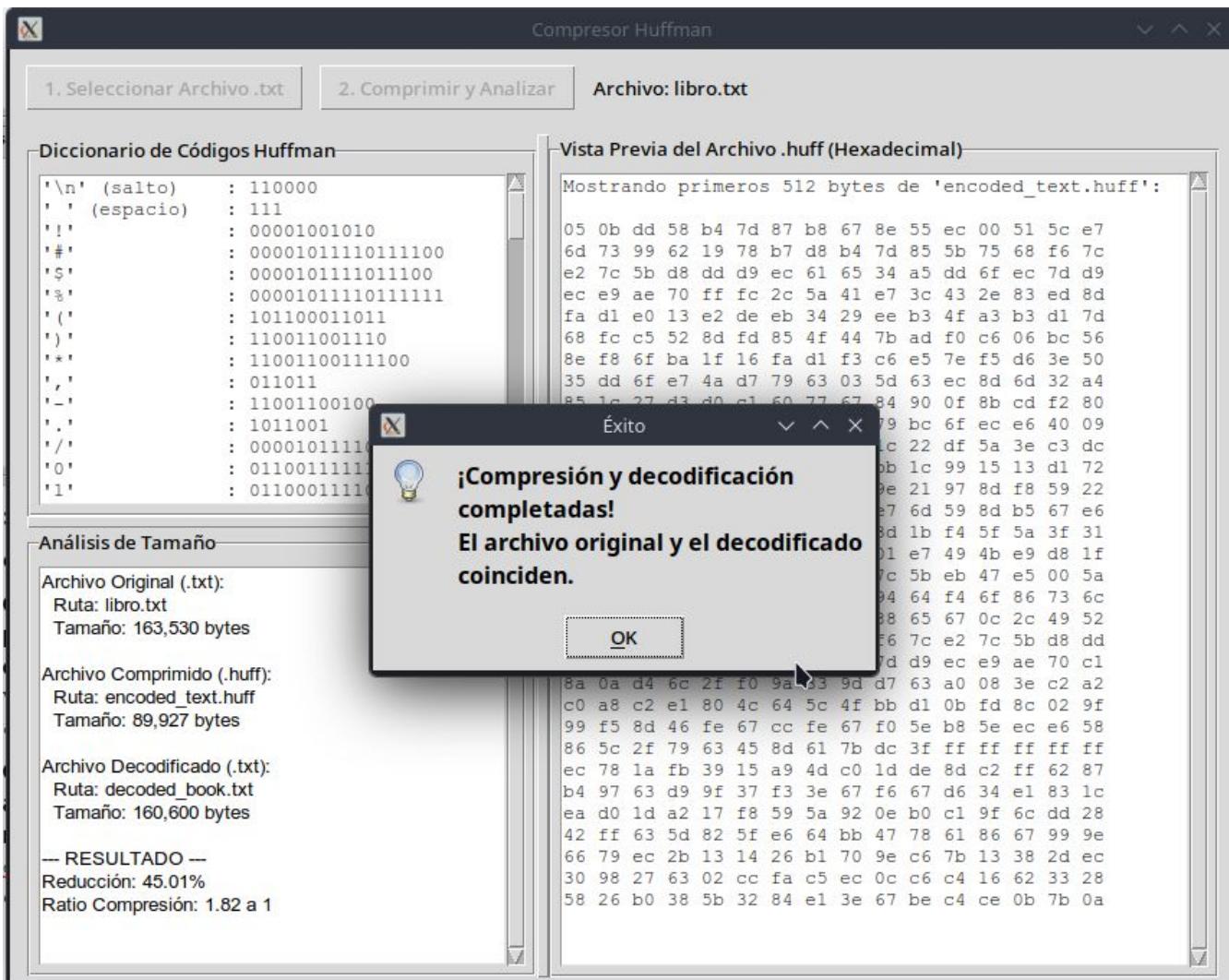
### Paso 3: Guardado de datos

La aplicación te guiará para guardar los dos archivos que generará.

- 1. Guardar el Comprimido (.huff):** El archivo comprimido se guardará automáticamente en la carpeta donde se encuentre el archivo `main.py` del programa con el nombre “`encoded_text.huff`”
- 2. Guardar el Decodificado (.txt):** Inmediatamente después, la aplicación leerá el archivo `.huff` que acabas de guardar y lo decodificará. Guardará los resultados en un archivo llamado “`decoded_book.txt`” en la ruta del programa `main.py`.

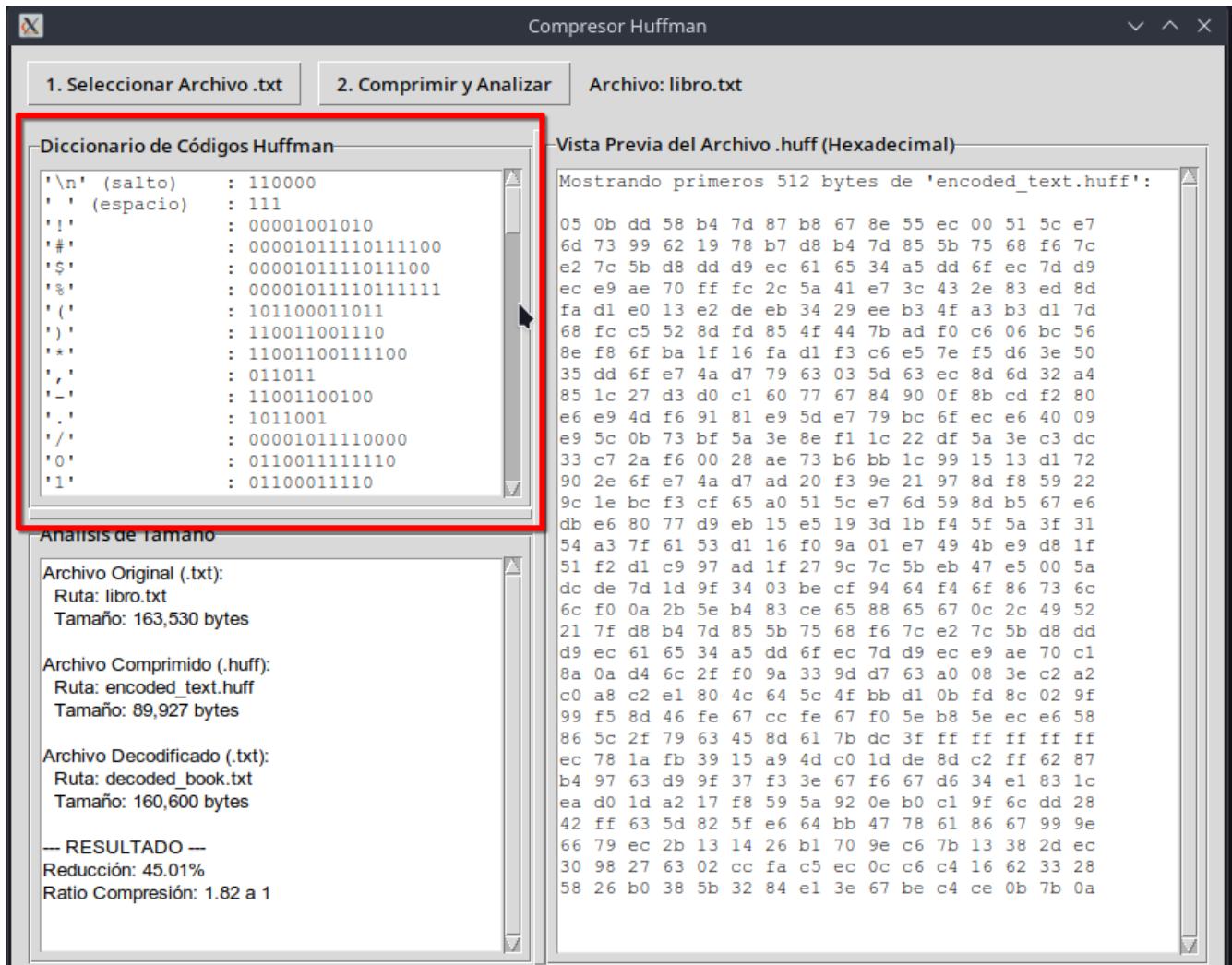
### 2. Revisar los Resultados

Una vez completado el proceso, la aplicación te mostrará los resultados de la compresión y la verificación. Ademas, primero se verá una ventana que avisa que el proceso se llevo a cabo con éxito.



## Códigos de Huffman

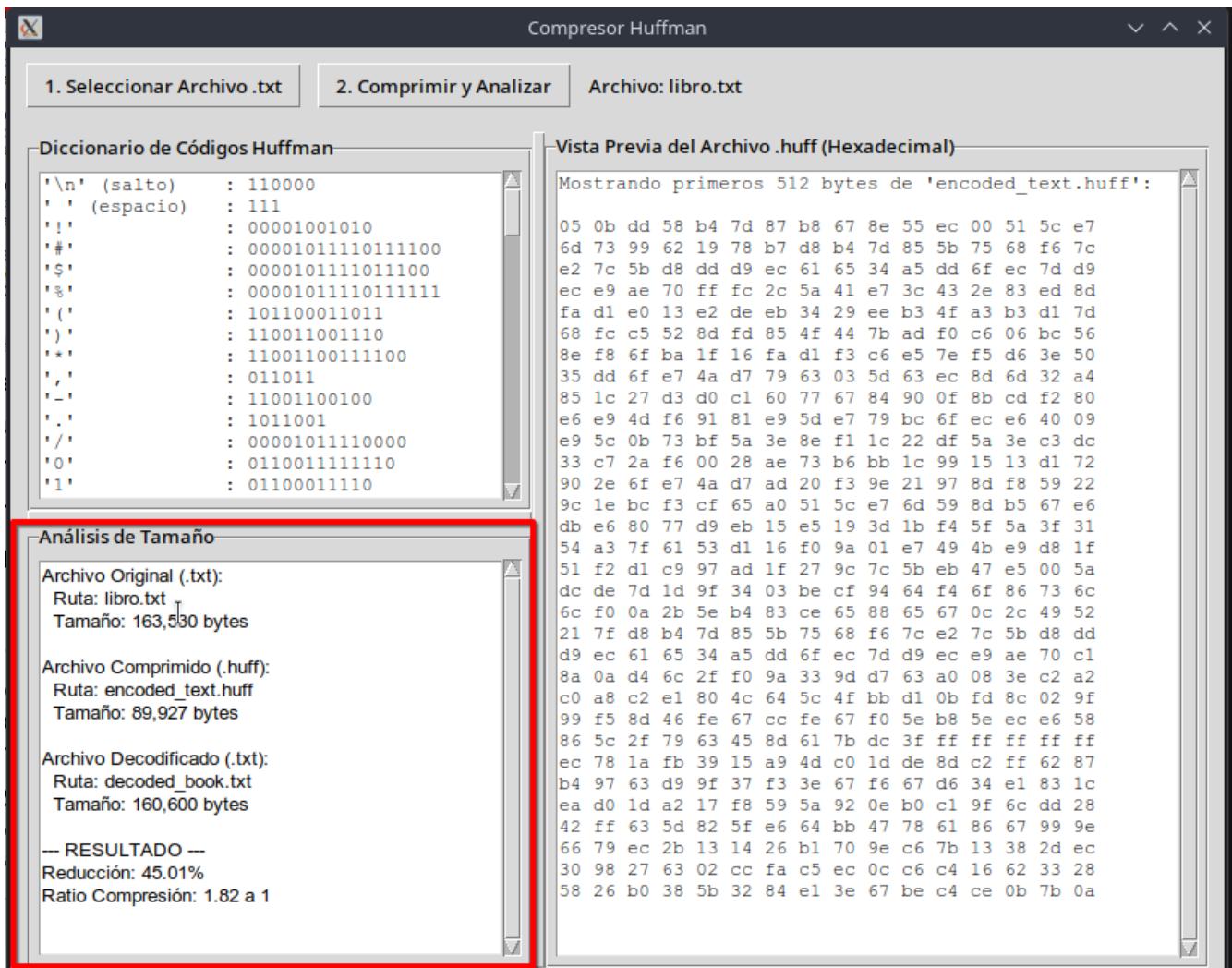
La interfaz mostrará los códigos binarios asignados a cada carácter. Los caracteres más comunes (como '' o 'e') tendrán códigos más cortos.



## Estadísticas de Compresión

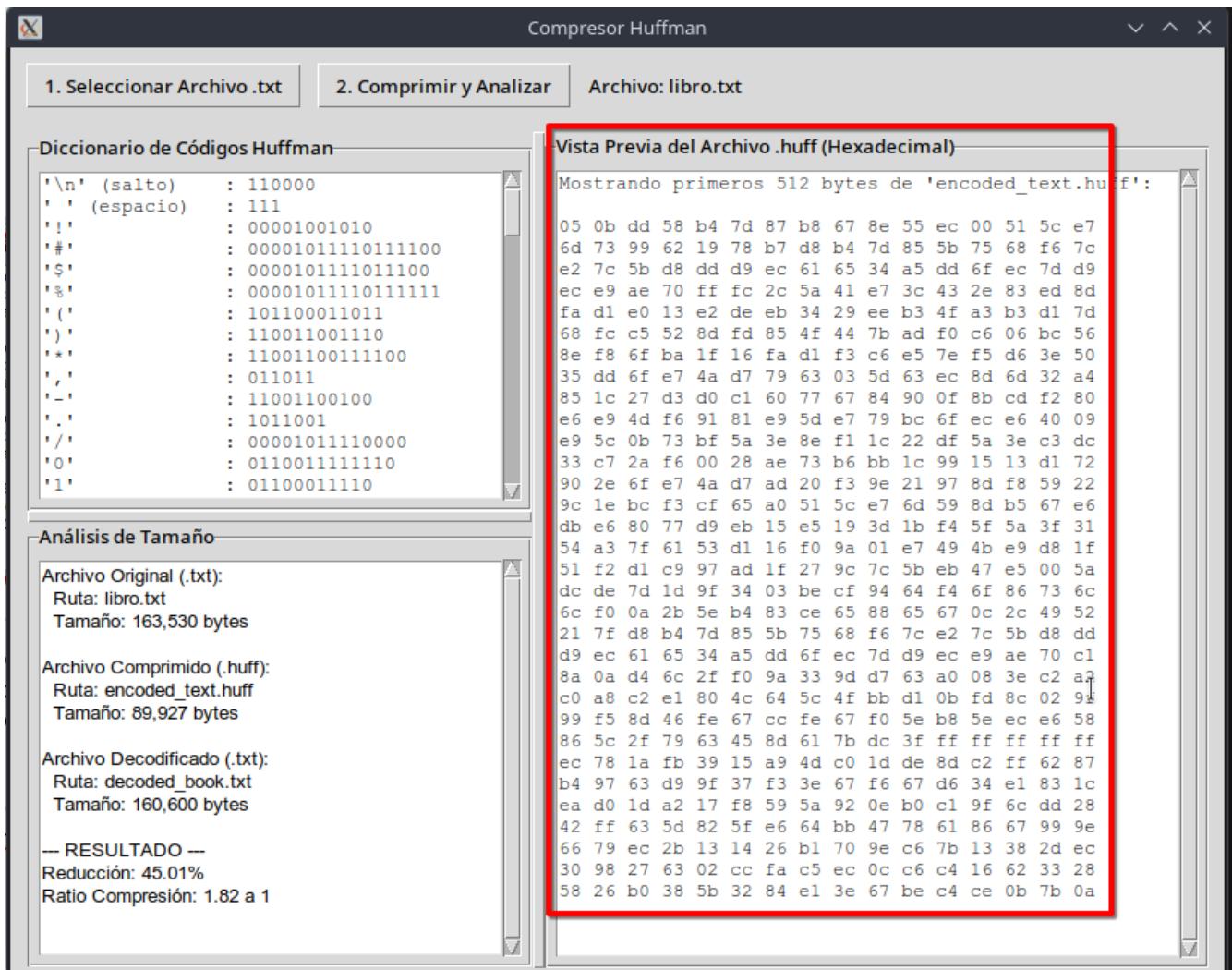
Verás un resumen del ahorro de espacio logrado:

- Tamaño Original:** El tamaño del archivo original en bits (ej. 8000 bits).
- Tamaño Comprimido:** El tamaño del nuevo archivo .huff en bits (ej. 4500 bits).
- Reducción:** El porcentaje de espacio ahorrado.



## Verificación

Finalmente, la aplicación te dará el resultado más importante: la visualización del nuevo archivo .huff generado. Sin embargo lo hará de una manera peculiar, podrás observar los primeros 512 bytes generados de manera hexadecimal.



¡Y eso es todo! Has comprimido y verificado un archivo exitosamente usando la codificación de Huffman.