

MANUAL DE USUARIO

COMPRESOR DE ARCHIVOS

V1.4.0



11/2025

Contenido

Introducción	3
Objetivo.....	3
Perfil de Usuario	3
Funciones del programa	4
Explicación del programa.....	4
Ejecución del programa	5
Compresión	7
Descompresión	8
Aclaraciones	8

Introducción

El programa compresor de archivos es un programa creado para la revisión, estandarización y simplificación de documentos de tipo texto (como los .txt) cuya finalidad es crear un archivo en donde solo se almacenen los binarios (un archivo .bin) en donde se traduce el documento a una versión mas simplificada del archivo original mas liviana para el computador.

Mediante una ejecución por terminal el usuario será capaz de ingresar el nombre del archivo que requiere comprimir y posteriormente recibir directamente el archivo .bin generado por el programa.

Objetivo

El objetivo del programa es permitir demostrar como mediante la utilización de distintas estructuras tales como son los árboles, las matrices dispersas y los diccionarios se pueden crear aplicaciones tan básicas y necesarias tales como son un compresor de archivos.

Adicionalmente el propósito de esta guía es el de guiar al usuario mediante el proceso de ejecución del programa para que pueda ejecutarlo sin complicaciones mayores y logre entender el paso a paso de como hacer un uso correcto de este programa.

Perfil de Usuario

Esta guía va dirigida a usuario básicos, común y corriente asumiendo que poseen conocimientos limitados con respecto a la lógica detrás del programa y únicamente requieren de la función principal del programa

Los conocimientos previos no son requeridos para este programa, sin embargo, para su correcta ejecución y entendimiento técnico es preferible conocer lo básico de estructuras de programacion.

Funciones del programa

(Esta primera parte es más teórica para aquellos interesados en conocer más a profundidad la implementación del código y maneja términos más técnicos de los esperados, en caso de no ser de su interés por favor avance a la siguiente hoja del manual donde se encontrará la sección de ejecución del programa)

Explicación del programa

El programa consiste en un código que recibe archivos de tipo texto (.txt) y los retorna como archivos de tipo binario (.bin) internamente el programa estandariza todo el documento a un mismo tipo de empaquetado de bits para facilidad de interpretar y simplificar los caracteres para el programa, posterior a esto usamos una simplificación del numero de bits a un numero mas simplificado que referencia este carácter, se crea una matriz donde se almacenan estos datos y se determina cual es el dato mas frecuente dentro de esta matriz.

Posteriormente, tras la creación de esta matriz generamos una nueva matriz en donde se indica el valor simplificado de cada carácter y se almacena junto con su fila y columna en una nueva matriz que será entregada para su proceso de simplificación.

Esta nueva matriz generada recibirá los datos y los procesará mediante el uso de un árbol y el algoritmo de Hoffman para su simplificación en un archivo de menor tamaño, generará un diccionario que referencia cada carácter con su binario correspondiente después de la codificación.

Finalmente, se genera un archivo binario (.bin) en donde viene inmerso: el archivo de texto simplificado a binario, el diccionario de referencia para su posterior codificación, las dimensiones originales del archivo fuente y el carácter mas usado para su referenciación en los espacios donde este ausente de caracteres el documento en su posterior decodificación.

Ejecución del programa

Para la ejecución del programa inicialmente es importante tener en cuenta la ubicación del programa.

Para corroborar la ubicación del programa ubíquelo dentro de su computadora y use clic derecho y oprima la opción de copiar ruta de acceso.

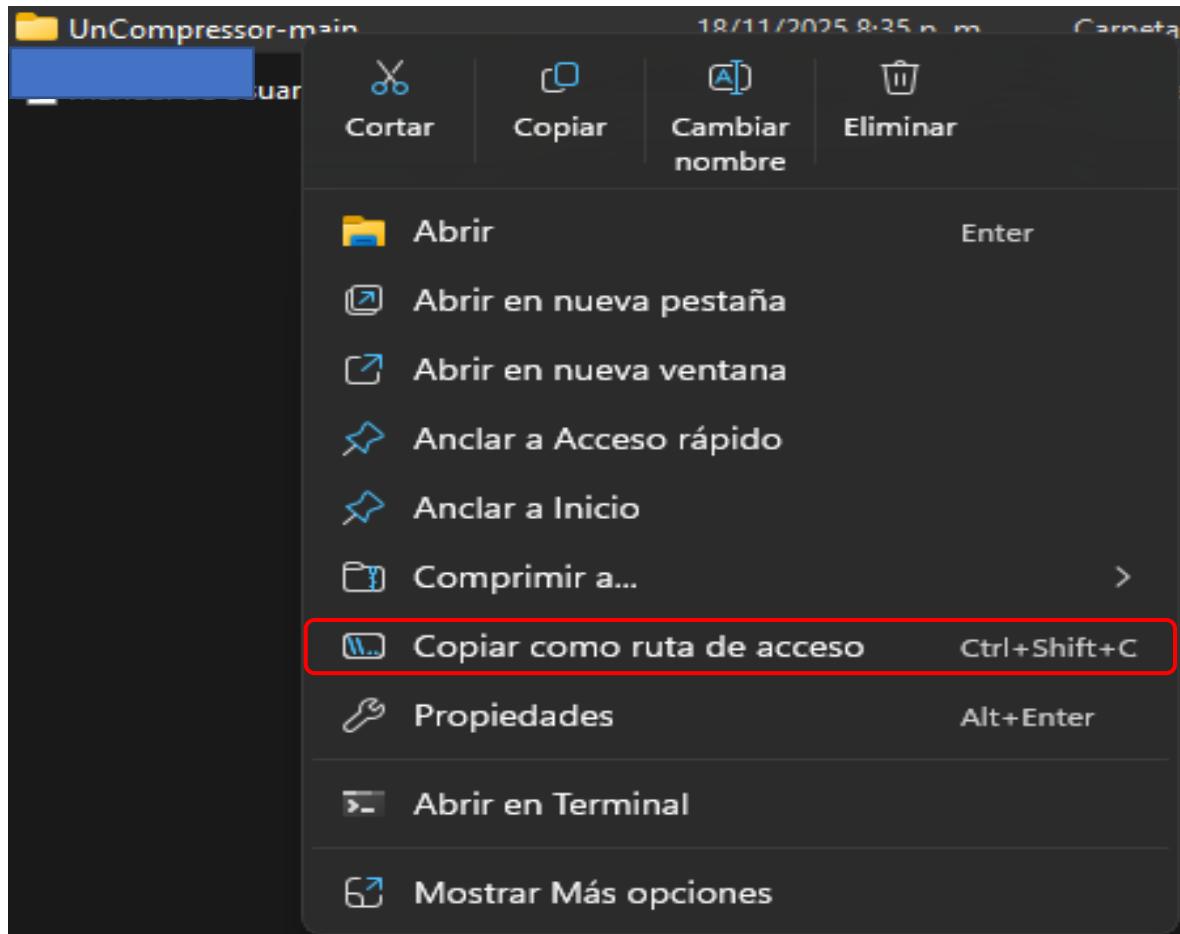


Imagen 1. Imagen guía de cómo obtener la ruta de acceso

Ya con la ubicación referenciada se procede a abrir el intérprete de C++ dentro del computador para poder ejecutarlo y se ejecuta el siguiente comando:

```
Cd "ruta_copiada_hasta_la_carpeta_en_donde_se_localiza_el_programa"
```

```
$ cd "ruta/UNCOMPRESSOR-MAIN"
```

Imagen 2. Imagen guía de como insertar la ruta de acceso en la terminal

Posterior a que muestre efectivamente la ruta del archivo en donde se ubique nuestra carpeta realizamos los siguientes comandos en su correspondiente orden:

- make

(este comando permite al makefile que tiene la información del programa indicar que compile todas las carpetas del programa y las ejecute correctamente al utilizar el main)

- make run

(este comando es el encargado de realizar la ejecución del main del programa)

Aquí se muestra cómo se debería de estar viendo al ejecutar estos comandos

Con la primera ejecución del make

```
$ make
Compilando main.cpp...
g++ -Wall -std=c++17 -g -Ilib/huffman/include -Ilib/lector/include -Ilib/dictionary/include -c src/main.cpp
-o build/main.o
Compilando librería: Decoder.cpp...
g++ -Wall -std=c++17 -g -Ilib/huffman/include -Ilib/lector/include -Ilib/dictionary/include -c lib/dictionary/src/Decoder.cpp -o build/Decoder.o
Compilando librería: Dictionary.cpp...
g++ -Wall -std=c++17 -g -Ilib/huffman/include -Ilib/lector/include -Ilib/dictionary/include -c lib/dictionary/src/src/Dictionary.cpp -o build/Dictionary.o
Compilando librería: HuffmanNode.cpp...
g++ -Wall -std=c++17 -g -Ilib/huffman/include -Ilib/lector/include -Ilib/dictionary/include -c lib/huffman/src/HuffmanNode.cpp -o build/HuffmanNode.o
Compilando librería: HuffmanTree.cpp...
g++ -Wall -std=c++17 -g -Ilib/huffman/include -Ilib/lector/include -Ilib/dictionary/include -c lib/huffman/src/HuffmanTree.cpp -o build/HuffmanTree.o
Compilando librería: MatrixHuffman.cpp...
g++ -Wall -std=c++17 -g -Ilib/huffman/include -Ilib/lector/include -Ilib/dictionary/include -c lib/huffman/src/MatrixHuffman.cpp -o build/MatrixHuffman.o
Compilando librería: lector.cpp...
g++ -Wall -std=c++17 -g -Ilib/huffman/include -Ilib/lector/include -Ilib/dictionary/include -c lib/lector/src/lector.cpp -o build/lector.o
Enlazando ejecutable...
g++ build/main.o build/Decoder.o build/Dictionary.o build/HuffmanNode.o build/HuffmanTree.o build/MatrixHuffman.o build/lector.o -o build/uncompressor
```

Imagen 3. Imagen guía de creación de los .build del programa

Posterior a esta ejecución el programa ya queda con el archivo build que inicializa estas librerías cuando se ejecuta el main

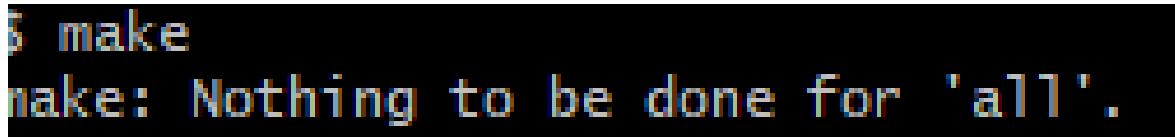


Imagen 4. Imagen guía texto confirmación de creación del ejecutable

Y se realiza la ejecución del main mediante el comando make run para usar le programa

```
$ make run
$ Ejecutando UnCompressor...
```

Imagen 5. Imagen guía de ejecución del programa

Al iniciar el programa aparecerá un simple menú que permite seleccionar entre las 2 funciones de nuestro programa

```
Selecione una opcion:
1) Comprimir texto
2) Descomprimir binario
Opcion [1]: |
```

Imagen 6. Imagen guía del menu

Compresión

Al ejecutarse la opción 1 aparecerá este texto en donde pide indicar la ruta del archivo a comprimir, en este caso del programa por facilidades de ejecución se recomienda que el archivo de texto se encuentre dentro de la carpeta del programa.

Se nombra la ruta y se utiliza entre para ejecutar el programa.

```
Compresando matriz.txt a matriz_comprimida.bin
--- COMPRESOR HUFFMAN PARA MATRICES DISPERSAS ---
Ingrese la ruta del archivo de texto a procesar: texto.txt
[INFO] Cargando y normalizando texto...
[INFO] Matriz generada: 546x57 con 12200 elementos no vacios.
--- Iniciando Codificacion Huffman ---
Valor de fondo (mas frecuente): '32'
[INFO] Elementos dispersos finales a comprimir: 12200
[BINARIO] Archivo optimizado generado: matriz_comprimida.bin

--- PROCESO TERMINADO CON EXITO ---
1. Archivo generado: matriz_comprimida.bin
2. Simbolos unicos en diccionario: 32
3. Codigo asignado al fondo ('32'): 1
```

Imagen 7. Imagen de ejecución del proceso de compresión

Al ejecutarse este programa con una ruta valida mostrara información general del proceso como los elementos internos del arreglo, las dimensiones de la matriz, el elemento más común dentro del arreglo y cuando se esté ejecutando el algoritmo de Hoffman.

Cuando aparezca el texto de proceso terminado vera que dentro de la misma carpeta se genero un nuevo archivo .bin donde se habrá almacenado el archivo comprimido de forma efectiva.

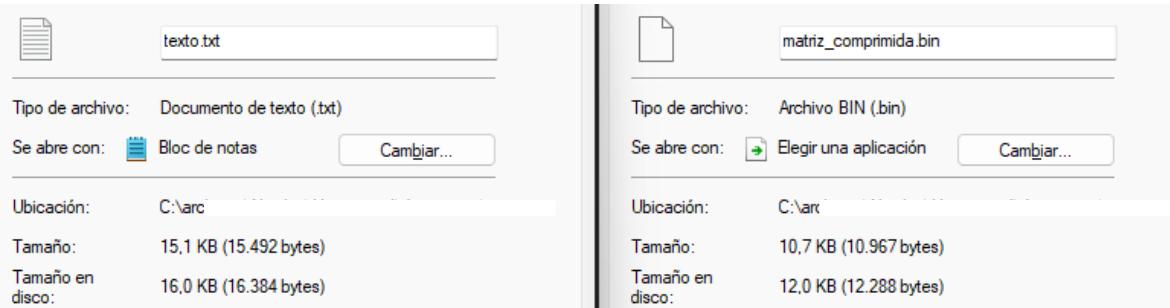


Imagen 8. Imagen comparativa de tamaños del archivo

Descompresión

Al seleccionar la opción numero 2, similar a el menú de compresión pedirá que le entreguemos un archivo para descomprimir, sin embargo en este caso será un archivo ya comprimido en .bin, por facilidades de este manual el archivo descompresor ya se encuentra dentro del mismo programa bajo el nombre de “matriz_comprimida.bin”

```
== DECODIFICADOR HUFFMAN PARA MATRICES DISPERSAS ==
Ingrese la ruta del archivo .bin a decodificar [matriz_comprimida.bin]: matriz_comprimida.bin
```

Imagen 9. Imagen guía de selección de archivo, menú de descompresión de archivos

Posteriormente el programa pedirá al usuario que asigne un nombre al nuevo archivo ya descomprimido, en este caso se mantendrá bajo el nombre predeterminado que otorga el programa “matriz_recuperada.txt”

```
Ingrese la ruta del archivo de texto donde guardar la salida [matriz_recuperada.txt]: matriz_recuperada.txt
[INFO] Decodificando 'matriz_comprimida.bin'...
[INFO] Matriz restaurada y guardada en 'matriz_recuperada.txt'.
```

Imagen 10. Imagen guía de nombramiento y descompresión del archivo

Cuando retorne los diálogos de decodificando y matriz recuperada el archivo se encontrará disponible en la misma carpeta del programa bajo el nombre que se le asignó anteriormente

Aclaraciones

El programa solo esta diseñado para archivos de texto .txt, no está disponible para otros tipos de formato además de texto como audio o video.

El programa funciona mediante una lógica de repetición, mientras mas extenso y repetitivo sean los caracteres más compactos será, sin embargo, con archivos de texto muy pequeños no será efectivo ya que generara archivos mas pesados que el original