



IC2001 Estructuras de Datos

Grupo 2

Jose Pablo Aguero Mora 2021126372

Katerine Guzmán Flores 2019390523

Ficha de revisión – Proyecto 3A

I Semestre 2022

Ficha de revisión

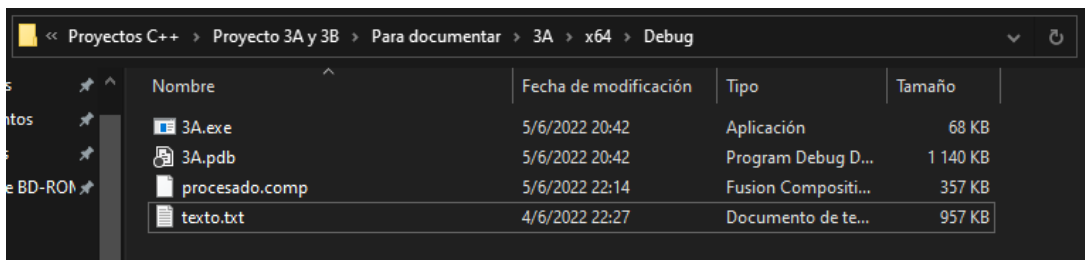
En este documento se va a detallar cada una de las funcionalidades que se aplicaron al proyecto según las especificaciones brindadas al inicio del mismo, con la meta de visualizar cuáles objetivos se cumplieron.

Funcionalidades incluidas:

- Aplicar un algoritmo de compresión Huffman. En este caso se asignan códigos a cada uno de los caracteres de un archivo. La compresión se consigue cuando se asignan códigos más cortos a los caracteres que tienen más porcentaje de aparición.

```
3 archivos      2 216 079 bytes
2 dirs  660 247 293 952 bytes libres

D:\Proyectos C++\Proyecto 3A y 3B\Para documentar\3A\x64\Debug>3A.exe e texto.txt procesado.comp
Proceso de compresion finalizado
D:\Proyectos C++\Proyecto 3A y 3B\Para documentar\3A\x64\Debug>
```



Nombre	Fecha de modificación	Tipo	Tamaño
3A.exe	5/6/2022 20:42	Aplicación	68 KB
3A.pdb	5/6/2022 20:42	Program Debug D...	1 140 KB
procesado.comp	5/6/2022 22:14	Fusion Compositi...	357 KB
texto.txt	4/6/2022 22:27	Documento de te...	957 KB

- De igual forma realizar una descompresión de archivos usando algoritmos de Huffman. De esta forma es necesario conocer el código asignado a cada carácter, así como su longitud en bits.

- Manejo de bits para agrupar repeticiones de caracteres.

```
/* Leer datos comprimidos y extraer al fichero de salida */
bits = 0;
fopen_s(&fs, argv[3], "w");
/* Lee los primeros cuatro bytes en la doble palabra bits */
fread(&a, sizeof(char), 1, fe);
bits |= a;
bits <= 8;
fread(&a, sizeof(char), 1, fe);
bits |= a;
bits <= 8;
fread(&a, sizeof(char), 1, fe);
bits |= a;
bits <= 8;
fread(&a, sizeof(char), 1, fe);
bits |= a;
j = 0; /* Cada 8 bits leemos otro byte */
q = Arbol;
```

- Es posible ejecutar el programa desde la línea de comandos con un sistema de instrucción definido: debe iniciar indicando el nombre del archivo ejecutable, en este caso "3A.exe", luego se decide el modo a utilizar, es decir: si desea comprimir entonces escribe una "e" y si desea descomprimir escribe una "d". Finalmente se debe escribir el nombre del archivo a procesar (junto con su extensión) y un nombre deseado para el nuevo archivo (este puede ser el que sea, al igual que su extensión). Cada uno de estos parámetros se debe separar con un espacio para que sea reconocido por el programa.

```
D:\Proyectos C++\Proyecto 3A y 3B\Para documentar\3A\x64\Debug>3A.exe e texto.txt procesado.comp
Proceso de compresion finalizado
D:\Proyectos C++\Proyecto 3A y 3B\Para documentar\3A\x64\Debug>3A.exe d procesado.comp nuevo-texto.txt
Proceso de descompresion finalizado
D:\Proyectos C++\Proyecto 3A y 3B\Para documentar\3A\x64\Debug>
```

Funcionalidades no incluidas:

- El sistema no comprime ciertos tipos de archivos (extensiones).

En resumen:

Indicación:	Realizado	No realizado
Aplicar un algoritmo de compresión Huffman. En este caso se asignan códigos a cada uno de los caracteres de un archivo. La compresión se consigue cuando se asignan códigos más cortos a los caracteres que tienen más porcentaje de aparición.	X	
De igual forma realizar una descompresión de archivos usando algoritmos de Huffman. De esta forma es necesario conocer el código asignado a cada carácter, así como su longitud en bits.	X	
La verificación de la compresión es viable al comparar los pesos del archivo original y el archivo comprimido.	X	
El contenido de ambos archivos es verificable, se puede confirmar la reconstrucción.	X	
El sistema no comprime ciertos tipos de archivos (extensiones).		X
Manejo de bits para agrupar repeticiones de caracteres.	X	
Es posible ejecutar el programa desde la línea de comandos con un sistema de instrucción definido.	X	