**Complejidad espacial**

Analisis de la complejidad temporal

Como podemos ver en el código fuente los dos programas estean unidos en el main y se separan por medio de un if, asi que para cada uno empesaremos desde los bloques try-cath que leen los txt y tomaremos solo los elementos de tamaño mayor a 1.

**Codificación huffman**

Ordenar()

Arreglo tamaño n

**ordenar = n**

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

buscarPosicion()

arreglo tamaño n

árbol tamaño m

**buscarPosicion() = n +m**

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

recorridoArbol()

2 listas tamaño n

árbol tamaño m

**recorridoArbol() = 2n + m**

**/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/**

codificacionHuffman()

lista tamaño n

buscarPosicion() = n + m

recorridoArbol() = 2n + m

**codificacionHuffman() = 3n + 2m**

**/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/**

Comprimir()

Lista tamaño n

Texto tamaño n

Texto tamaño m

**Comprimir() = 2n + m**

**/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/**

contarFrec()

texto tamaño n

lista tamaño m

**contarFrec() = n + m**

**/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/**

Main()

Texto tamaño n

2 Listas tamaño m

contarFrec() = n + m

codificacionHuffman() = 2n + m

Comprimir() = n + m

**Main() = n + 2m +n +m +2n +m +n+m = 5n + 5m**

**Decodificación huffman**

decodificacionHuffman()

texto tamaño n

texto tamaño m

lista tamaño m

**decodificacionHuffman() = n + 2m**

**/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/**

Main()

Lista tamaño n

Texto tamaño m

Texto tamaño n

decodificacionHuffman() = n +2m

**Main() =2n +m +n +2m = 3n + 3m**

**La complejidad de codificaciónHuffman es O(n) = 5n + 5m**

**La complejidad de decodificacionHuffman es O(n) = 3n + 3m**