

Albert Folch Garcia
Xavi Moreno Liceras

Exercici AvCont-4b: LZ77 text

1.- Haced las modificaciones necesarias en el programa anterior para que sea capaz de leer ficheros de texto, generar el código ASCII (de 8 bits) de cada carácter del texto y devolverlo como una String binaria procesable por vuestros métodos de compresión /descompresión en LZ-77. Conseguid también que el programa calcule el tiempo invertido en comprimir los datos. Comprobad el correcto funcionamiento del programa; es decir, que es capaz de comprimir texto en LZ-77 y de recuperarlo tras la descompresión. Ayuda: Utilizad los métodos de la clase `txtReader` que se os proporciona Ayuda: Medid el tiempo mediante `System.nanoTime()`. Ayuda: Se os proporcionan los archivos “hamlet_short.txt” y “quijote_short.txt”, que contienen dos fragmentos de texto en inglés y en español, respectivamente.

Ejecutar el lz77-text con el siguiente comando: `python lz77-text.py`
Veremos que nos saldrá:

Hint:

- * compress: `python lz77-text.py my/path/myfile.txt /my/path/namecompressfile`
- * decompress: `python lz77-text.py my/path/namecompressfile`

Por ejemplo podemos comprimir poniendo:

```
python lz77-text.py resource/hamlet_short.txt hamlet
```

Y para descomprimir el fichero comprimido:

```
python lz77-text.py hamlet
```

Esto te creará un fichero llamado: ‘hamlet.txt’.

Si se quiere cambiar el *Mdes* i el *Ment*, se puede cambiar desde `lz77-text.py`, aunque por defecto ya está puesto un *Mdes* i *Ment* más óptimo posible.

2.- Comprimid el archivo “hamlet_short.txt” con distintos valores de Mdes y Ment entre 4 y 4096 y analizad el factor de compresión y el tiempo invertido para conseguirlo. ¿Cuál es el mejor factor de compresión que obtenéis y con qué valores de Mdes y Ment? ¿Cómo varia (cómo escala) el tiempo de cálculo necesario al aumentar Mdes y Ment? ¿Qué combinación de Mdes y Ment elegiríais?

Los factores de compresión que obtenemos son los siguientes:

Mdes	Ment	Tiempo compresión	Factor
2048	32	2.03 segons	1.0995 : 1
2048	64	5.94 segons	1.1108 : 1
2048	128	13.25 segons	1.0982 : 1
2048	256	30.15 segons	1.0465 : 1
2048	512	59.54 segons	1.0124 : 1
4096	32	1.62 segons	1.1237 : 1
4096	64	5.03 segons	1.1842 : 1
4096	128	11.78 segons	1.1983 : 1
4096	256	27.19 segons	1.1744 : 1
4096	512	54.86 segons	1.1508 : 1
8192	32	0.21 segons	0.9998 : 1
8192	64	0.77 segons	1.0012 : 1
8192	128	1.41 segons	0.9996 : 1
8192	256	2.21 segons	0.9979 : 1

Conforme aumentamos Mdes, el tiempo de compresión disminuye. En cambio, si aumentamos Ment, el tiempo también aumenta.

La combinación de Mdes y Ment que cogeríamos sería:

- Mdes: 4096
- Ment: 128

3.- Comprimid ahora el archivo “quijote_short.txt” y analizad para qué combinación de Mdes y Ment se obtiene el mejor factor de compresión. ¿Es el mismo que en el caso anterior? Proponed varias razones que expliquen esta diferencia.

Mdes	Ment	Tiempo compresión	Factor
2048	32	1.8001 segons	1.1237 : 1
2048	64	5.7815 segons	1.1030 : 1
2048	128	13.7527 segons	1.0578 : 1
2048	256	30.1830 segons	1.0126 : 1
2048	512	63.0708 segons	0.9753 : 1
4096	32	1.8335 segons	1.1114 : 1
4096	64	6.0611 segons	1.1101 : 1
4096	128	15.0765 segons	1.0841 : 1
4096	256	33.7844 segons	1.0546 : 1
4096	512	68.0233 segons	1.0329 : 1
8192	32	0.0346 segons	1.0010 : 1

Conforme aumentamos Mdes, el tiempo de compresión disminuye. En cambio, si aumentamos Ment, el tiempo también aumenta.

La combinación de Mdes y Ment que cogeríamos sería:

- Mdes: 2048
- Ment: 32