

# Assignment #3 109503003

## 陳柏禎 通訊三

### Huffman coding

```
(base) paul@paulMacBook-m1 huffman-main % ./test -i paper.txt -o complete -c  
花費時間： 0.009515
```

使用霍夫曼編碼對一論文 (paper.txt) 壓縮的時間為0.009515

```
(base) paul@paulMacBook-m1 huffman-main % ./test -i complete -o paper.txt -d  
花費時間： 0.010940
```

使用霍夫曼編碼對一壓縮檔 (complete) 解壓縮的時間為0.010940

### Arithmetic coding

```
J00D00M00$0jZ000\0I0WY00i0;cC0H000'zW00z_0Rj0u0E000/e2[00|0^0"00`'3000000G00g0000L{00:u0E6%w0/0S000?0000@  
0Px0Sx,`/}00o00}0V$]ixriL00 t;0 0p004N0bC'0W0$`0dn600090  
#=003000sp00SE`:081070000000 00 000056g 00m00P'20g\B<u%000\";01-0000b 0000g00h00GZ#:dzd00S00  
00]0=U00d,1q0t-J0 0000g00h00GZ#:dzd00S00  
M00}0,00XPaj(08001q"7000m0k001D00~00b0x500p0v00{0>F>0!.06000000u00c0I00ijNU)n0L+W0~tL000q0B0@0c00n0G0W00Q0  
nLb P00000g002rH IG#0H30000)0zS8gS000w300z q05  
S0900000^0000+0F40鑄00U0P-[6 <0S0j001m00<3  
.0c0;0!偷/!0000000K0y00|D0000V^0h000,8m0000_0?0@+F000e0+3000?000_080X000S0%a0fv;[]j0m  
0;00Lni0C0000e00x0000[0#s0000te000#0$0|000Q%`00000080Tx000v00y90Yy0鑄a1x>1G|0G 0+00S0N0700  
花費時間： 0.045103
```

使用算術編碼對一論文 (paper.txt) 壓縮的時間為0.045103

```
(B/0kH03000wV00 <+ Z 000~00q00S00ea0I000000Z1#0t0d[]00  
00`008)*: ?b0o0+00M00<kJ0I0S0ir"0000]m000n00+w0Y<s~e03tFF0  
s 00004鑄 b00V000Z0LG)>0.T90000Z000w0I&y0u  
o00000G00V0000Q000Q0D0L@0T0  
0Y r500#+R8,D0'}@00000d00v0000,0000{0v0{0>0C0Q07000'++00)000  
~0e0000000 00TI000U|0H0$}\0M00FF t0h0000S00000M0Xw000<nv8p賄PG0x00Q\02F1q000)0!b000w0 0[]00C00C"C*00g0U&0  
0BG00B&00n00B'q0c03M-0x?  
00000500E0;000  
0K000D  
0000e000Z00000KX000QZ00*7lw0L00"00Ej}0q0t0B|0F000/!bM0002|H*kv]-sj400R0yp00C.^0x000+0; '*e007~0A0!0j0g00  
花費時間： 0.026808
```

使用算術編碼對一壓縮檔 (complete) 解壓縮的時間為0.026808

# 分析

就實驗結果而言，Huffman coding 壓縮以及解壓縮時間都較Arithmetic coding短，並且Huffman coding 的壓縮時間來得比解壓縮時間小很多。Arithmetic coding則是相反，解壓縮時間較壓縮時間快。

壓縮率：

- Huffman coding 可以達到接近最佳的壓縮率，特別是在資料中有一些字元出現頻率很高的情況下。
- Arithmetic coding 可以達到理論最佳的壓縮率，因為它可以更精確地儲存每個字元的出現機率

複雜度：

- Huffman coding 實現較簡單
- Arithmetic coding 實現較複雜

解壓縮時間：

- Huffman coding 解壓縮時間為 $O(n)$ ,因為只需要從根節點開始走下去就可以了
- Arithmetic coding 解壓縮時間為 $O(\log n)$ ,需要對每個字元進行二分搜尋

結論：

Huffman coding比較適合頻率分布不均的資料壓縮，而Arithmetic coding比較適合頻率分布平均的資料壓縮