(2)

(a)

$$\text{entrophy} = \sum_i P(S_i) \ln \frac{1}{P(S_i)}$$

$$= \sum_{n=1}^{80000} (1 - \exp(-0.002n)) \cdot \ln \frac{1}{1 - \exp(-0.002n)}$$

$= 7.21$

(b)

By Shannon Coding theory, $\underbrace{\frac{\text{entropy}}{\ln k}}_{10.4} \le$ mean of Huffman code $\le \underbrace{\frac{\text{entropy}}{\ln k}}_{11.04} + 1$

$\Rightarrow$ range of L should be $(1.04 \times 10^6,\ 1.104 \times 10^6)$

(c)

if Arithemetic coding has coding length b

$\Rightarrow \text{ceil}\left(N \cdot \frac{\text{entropy}}{\ln k}\right) \le b \le \text{floor}\left(N \cdot \frac{\text{entropy}}{\ln k} + \log_k 2 + 1\right)$

range of b = 1040848 ~ 1040849

(3)

$e^{j\theta} \cdot x = (\cos\theta + j\sin\theta)(a + jb) \Rightarrow \begin{bmatrix} c & -s \\ s & c \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix}$

$\Rightarrow \begin{bmatrix} c & -s \\ s & c \end{bmatrix} = \underbrace{\begin{bmatrix} c & c \\ c & c \end{bmatrix}}_{1\ \text{MUL}} + \underbrace{\begin{bmatrix} 0 & -s-c \\ s-c & 0 \end{bmatrix}}_{2\ \text{MULs}}$

if $s-c=0$ or $-s-c=0$ or $c=0$, then just 2MULs are needed.

$\Rightarrow \begin{matrix} s=c \\ s=-c \end{matrix} \Rightarrow \theta = \frac{\pi}{4} + \frac{\pi}{2}n \quad n=0,1,2,3$ #

$c=0 \Rightarrow \theta = \frac{\pi}{2}n \quad n=0,1,2,3$ #

(4)

for computing 1D FFT of N pts, we use divide & conquer the array into N/2, N/4 ..
until size of subarray = 1, and do O(N) operation for each subarray.

$\Rightarrow$ 1D DFT has time complexity = $O(N\log N)$

Similarly, for 3D DFT, we do 1D FFT along three axis, which is dividing $M\times N\times P$ array into $N\times P$ M pts array, $M\times P$ N pts array, $M\times N$ P pts array and implement FFT algorithm.

Eventually, we have time complexity $O(N\times P\times M\log M)$, $O(M\times P\times N\log N)$, $O(M\times N\times P\log P)$ when it comes to computing along each axis. That is, the total time complexity t_total =

$O(M\times N\times P(\log M+\log N+\log P)) = O(M\times N\times P\log(M\times N\times P))$ #

(5)

$$\begin{array}{c} \begin{array}{cccc} a & b & b & a \end{array} \\ \begin{bmatrix} b & a & -a & -b \\ b & -a & -a & b \\ a & -b & b & -a \end{bmatrix} \end{array}$$

$$\Rightarrow \begin{bmatrix} z_1[1] \\ z_1[4] \end{bmatrix} = \underbrace{\begin{pmatrix} a & b \\ a & -b \end{pmatrix}}_{A_1} \begin{bmatrix} x[1] \\ x[2] \end{bmatrix} \quad \begin{bmatrix} z_2[1] \\ z_2[2] \end{bmatrix} = \underbrace{\begin{pmatrix} b & a \\ b & -a \end{pmatrix}}_{A_3} \begin{bmatrix} x[1] \\ x[2] \end{bmatrix}$$

$$\begin{bmatrix} z_1[2] \\ z_1[3] \end{bmatrix} = \underbrace{\begin{pmatrix} b & a \\ b & -a \end{pmatrix}}_{A_2} \begin{bmatrix} x[3] \\ x[4] \end{bmatrix} \quad \begin{bmatrix} z_2[1] \\ z_2[2] \end{bmatrix} = \underbrace{\begin{pmatrix} -a & -b \\ -a & b \end{pmatrix}}_{A_4} \begin{bmatrix} x[3] \\ x[4] \end{bmatrix}$$

$X = z_1 + z_2$

$\Rightarrow A_1 = -A_4 \quad A_2 = A_3 \Rightarrow$ all case four

$\Rightarrow$ 4 ADDs + 4 × (3 MULs + 3 ADDs) = 16 ADDs, 12 MULs

(6)

(a)

$143 = 11 \times 13$

$11 \times 52 + 13 \times 40 = 1092$

(b)

13×15

$15 \times 52 + 13 \times 40 = 1300$

(c)

$196 = 14 \times 14$

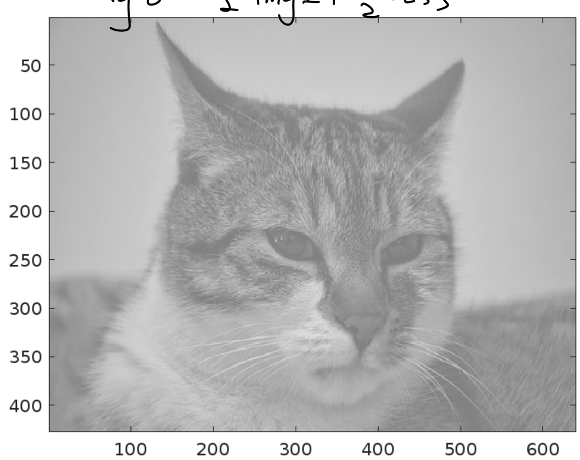
$14 \times 32 + 14 \times 32 = 896$

(1)

img 1

img 2

$\text{img}3 = \frac{1}{2}\text{img}1 + \frac{1}{2} \cdot 255$




```
>> ADSPpa4
SSIM between img1 and img2: -0.0089991
SSIM between img1 and img3: 0.68445
img1, img3 are more alike.
```

如同上課所展示的結果

SSIM 與肉眼所見的相似度較對齊

special question (3,8)

$$10 = (n_1 \cdot 3 + n_2 \cdot 7) \bmod 21$$

when $n_1 = 1 + 7k$, $n_2 = 1 + 3p$ $\quad k, p = 0, \pm 1, \pm 2 \cdots$

for $n_1, n_2 \in \{0, 1, 2 \cdots 20\}$ $\quad n_1 = n_2 = 1$ #