

**Multimedia Communications (INT-3305): Final exam - HK1 - 2017-2018**

*Thời gian:* 90 phút

*Không được sử dụng:* Internet

*Được sử dụng:* máy tính cầm tay, cheat sheet

*Tên:* .....

*MSSV:* .....

*Ngày sinh:* .....

*Chữ ký giám thị 1:* .....

*Chữ ký giám thị 2:* .....

**Bài 1**

Các phát biểu sau đúng hay sai:

1. Thuật toán nén LZW không làm giảm Entropy của dữ liệu đầu vào
2. Một encoder mã hoá các symbol thành các codeword (từ mã) như ở dưới.  
 $a=0010$   
 $b=0111$   
 $c=00$   
 $d=11$   
 Encoder này là không phải là một prefix code
3. Sampling (subsampling) có thể coi là một cách nén dữ liệu
4. Các hệ thống đảm bảo QoS theo kiểu IntServe sẽ có khả năng mở rộng (scalability) cao hơn các hệ thống theo kiểu DiffServe
5. Huffman, Arithmetic Coding, và LZW là các mã hoá Entropy coding
6. Entropy coding không khiến dữ liệu bị mất mát
7. Huffman, Arithmetic Coding, và LZW cần biết tần suất (xác suất) xuất hiện của từng symbol
8. Đánh dấu và phân loại gói tin là đủ để đảm bảo thực hiện được đảm bảo QoS ở lớp mạng
9. Các cách tiếp cận để giải quyết bài toán đảm bảo QoS là IntServe, DiffServe, và Laissez Faire
10. Mã hoá kênh (Channel coding) sẽ làm giảm lượng dữ liệu đi
11. Adaptive Playout Delay là một kỹ thuật đảm bảo QoS ở lớp mạng
12. Trong các ứng dụng Multimedia trên mạng Internet, tăng playout delay sẽ làm giảm mất mát gói tin.
13. Weighted Fair Queueing là một thuật toán có thể được áp dụng ở lớp mạng để đảm bảo QoS.
14. Mã sửa lỗi trước (Forward Error Correction - FEC) là một cách mã hoá nguồn (Source coding)
15. Giao thức TCP chạy trên mạng Long-fat (long-fat network) sẽ không đạt được hiệu quả cao

... xuất xuất hiện p(x), thì nội dung thông tin của nó (thông tin chứa) ...  
 ... giá trị biểu tượng, chỉ phụ thuộc vào xác suất xuất hiện ...  
 ... symbols trong alphabet của người:  $H(x) = \sum_{x \in X} p(x) \log_2 \frac{1}{p(x)}$  ...  
 ... short codes ...  
 ... (1,2) (1,4)  
 ... 3, 1)  
 ... symbol

**Bài 2**

Hãy encode chuỗi LALAL\* thành nhị phân sử dụng thuật toán LZW 9-bit.

Lưu ý:

- Ký tự \* là ký tự kết thúc, không cần mã hoá.
- Trình bày các bước mã hoá trung gian
- Trình bày chuỗi bit đầu ra thành từng khúc 9 bit sao cho dễ đọc

Gợi ý: Bảng mã ASCII có 256 ký tự. Trong đó 128 ký tự đầu tiên cho ở hình dưới

Dec	Hx	Oct	Char	Dec	Hx	Oct	Hlml	Chr	Dec	Hx	Oct	Hlml	Chr
0	0	000	NUL (null)	32	20	040	#32	Space	64	40	100	#64	@
1	1	001	SOF (start of heading)	33	21	041	#33	!	65	41	101	#65	A
2	2	002	STX (start of text)	34	22	042	#34	"	66	42	102	#66	B
3	3	003	ETX (end of text)	35	23	043	#35	#	67	43	103	#67	C
4	4	004	EOT (end of transmission)	36	24	044	#36	\$	68	44	104	#68	D
5	5	005	ENQ (enquiry)	37	25	045	#37	%	69	45	105	#69	E
6	6	006	ACK (acknowledge)	38	26	046	#38	&	70	46	106	#70	F
7	7	007	BEL (bell)	39	27	047	#39	'	71	47	107	#71	G
8	8	010	BS (backspace)	40	28	050	#40	(	72	48	110	#72	H
9	9	011	TAB (horizontal tab)	41	29	051	#41	)	73	49	111	#73	I
10	A	012	LF (NL line feed, new line)	42	2A	052	#42	*	74	4A	112	#74	J
11	B	013	VT (vertical tab)	43	2B	053	#43	+	75	4B	113	#75	K
12	C	014	FF (NP form feed, new page)	44	2C	054	#44	,	76	4C	114	#76	L
13	D	015	CR (carriage return)	45	2D	055	#45	-	77	4D	115	#77	M
14	E	016	SO (shift out)	46	2E	056	#46	.	78	4E	116	#78	N
15	F	017	SI (shift in)	47	2F	057	#47	/	79	4F	117	#79	O
16	10	020	DLE (data link escape)	48	30	060	#48	0	80	50	120	#80	P
17	11	021	DC1 (device control 1)	49	31	061	#49	1	81	51	121	#81	Q
18	12	022	DC2 (device control 2)	50	32	062	#50	2	82	52	122	#82	R
19	13	023	DC3 (device control 3)	51	33	063	#51	3	83	53	123	#83	S
20	14	024	DC4 (device control 4)	52	34	064	#52	4	84	54	124	#84	T
21	15	025	NAK (negative acknowledge)	53	35	065	#53	5	85	55	125	#85	U
22	16	026	SYN (synchronous idle)	54	36	066	#54	6	86	56	126	#86	V
23	17	027	ETB (end of trans. block)	55	37	067	#55	7	87	57	127	#87	W
24	18	030	CAN (cancel)	56	38	070	#56	8	88	58	130	#88	X
25	19	031	EM (end of medium)	57	39	071	#57	9	89	59	131	#89	Y
26	1A	032	SUB (substitute)	58	3A	072	#58	:	90	5A	132	#90	Z
27	1B	033	ESC (escape)	59	3B	073	#59	;	91	5B	133	#91	[
28	1C	034	FS (file separator)	60	3C	074	#60	<	92	5C	134	#92	\
29	1D	035	GS (group separator)	61	3D	075	#61	=	93	5D	135	#93	]
30	1E	036	RS (record separator)	62	3E	076	#62	>	94	5E	136	#94	^
31	1F	037	US (unit separator)	63	3F	077	#63	?	95	5F	137	#95	_

Source: www.LookupTables.com

**Bài 3**

Cho chuỗi dữ liệu đầu vào (hàng thứ 2)

vị trí	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
symbol	A	B	A	A	C	A	B	A	A	B	B	C	A	A	A	B	B	A	A	A

1. Hãy tính lượng tin của từng symbol A, B, C
2. Hãy tính entropy của chuỗi dữ liệu trên
3. Hãy vẽ cây mã hoá theo thuật toán mã hoá Huffman
4. Hãy cho biết mã Huffman tương ứng cho mỗi symbol A, B, C

Họ và tên thi: \_\_\_\_\_  
 Số báo danh: \_\_\_\_\_  
 Phòng thi: \_\_\_\_\_

• Symbol x với xác suất xuất hiện...  
 ... không phụ thuộc vào giá trị...  
 ... trung bình lượng tin đầu với mỗi sym...  
 Frequent symbols are encoded with...  
 ~004 => 9 12 5/4  
 ~ (3,1) (3,2) (4,7) (1,5) (2,6) ...  
 ~mv (lossless)  
 ~"h" (codeword)  
 ~hel. usually (0,1)  
 ~ = sym

**Bài 4**  
 Xét một token bucket với tham số  $r_{token} = 2000 \text{ tokens/s}$ , và bucket size  $b = 1000 \text{ tokens}$ . Kích thước hằng đợi đầu vào của token bucket là vô hạn.

1. Người ta cho 1 luồng dữ liệu chảy liên tục vào đầu vào của token bucket trên trong một thời gian dài, tốc độ dữ liệu đầu ra  $r_{out}$  của token bucket sẽ là bao nhiêu, nếu:

A. tốc độ của luồng dữ liệu đầu vào là  $r_{in} = 3000 \text{ packets/s}$   
 B. tốc độ của luồng dữ liệu đầu vào là  $r_{in} = 1500 \text{ packets/s}$

2. Nếu có 1 burst dữ liệu rộng 500ms chứa 5000 packets áp đến đầu vào của token bucket. Sau bao lâu toàn bộ 5000 packets sẽ được gửi toàn bộ đi. Giả sử rằng tại thời điểm bắt đầu của burst dữ liệu thì bucket đang chứa đầy  $b$  tokens.

$$\frac{5000}{2000 \text{ packet/s} + b} \cdot 10^9$$

$$\frac{r_{in}}{r_{out}} = \frac{r_{in} \cdot t + b}{r_{out} \cdot t}$$

$$r_{out} = \frac{r_{in} \cdot t + b}{t}$$

$$r_{out} = 10^9 \text{ pk/s} + b$$

**Bài 5**  
 Cho tín hiệu  $s(t)$  có dạng sóng

$$s(t) = 4 \times \sin(2\pi 8 \times 10^3 t) + 5.2 \times \cos(6\pi 8 \times 10^3 t)$$

1. Tần số lấy mẫu tối thiểu của tín hiệu  $f_{min}$  là bao nhiêu?  
 2. Giả sử tín hiệu  $s(t)$  được lấy mẫu với tần số  $f_{min}$ , sau đó từng mẫu được lượng tử hoá (quantize) 8 bit/mẫu. Hỏi tốc độ bit của luồng dữ liệu (sau khi số hoá tín hiệu  $s(t)$ ) là bao nhiêu?  
 3. Giả sử tín hiệu  $s(t)$  được lấy mẫu với tần số  $f_{min}$  rồi mã hoá DPCM. Cụ thể là, cả luồng được chia thành từng frame 1500ms. Với mỗi frame mẫu đầu tiên được lượng tử hoá thành 8bit/mẫu, các mẫu tiếp theo được mã hoá thành hiệu giữa mẫu hiện tại với mẫu trước đó rồi lượng tử hoá 4bit/mẫu. Hỏi, tốc độ bit của luồng dữ liệu DPCM là bao nhiêu?

$$8 \cdot \frac{1}{48} \cdot 10^3$$

$$\frac{8}{48} \cdot 10^3$$

$$\frac{10^9 t + b}{r_{out}}$$

88000 Hz 8 bit/mẫu  
 48 ms/mẫu

**Bài 6**  
 Người ta sử dụng một cơ chế mã hoá FEC cho một chuỗi các gói đầu vào. Cứ  $n$  gói dữ liệu đầu vào (gói tin gốc - original packet) được mã hoá ra  $k$  gói dữ liệu mã hoá theo công thức dưới

$$\mathbf{X}_n = \mathbf{A} \mathbf{x}_n$$

Trong đó:

- $\mathbf{x}$ : vector (cột) các gói tin gốc (original packets)
- $\mathbf{X}$ : vector (cột) các gói tin mã hoá (encoded packets)
- $\mathbf{A}$ : Ma trận mã hoá và

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix}$$

$$\mathbf{x}_1 \dots \mathbf{x}_6$$

$$\mathbf{X}_1 \dots \mathbf{X}_6$$

$$\mathbf{X}_6 = \mathbf{x}_1 + \mathbf{x}_5 \quad \mathbf{X}_2 + \mathbf{x}_5 + \mathbf{x}_6$$

Câu hỏi:

- Với cơ chế mã hoá ở trên  $n$  bằng bao nhiêu?  $k$  bằng bao nhiêu?
- Viết công thức tính các gói mã hoá  $\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_k$

