UNIVERSITY OF TORONTO

FACULTY OF ARTS AND SCIENCE

APRIL / MAY 2003 EXAMINATIONS

CSC 310F - INFORMATION THEORY

Duration - 2 hours

- **Open books and notes.**
- **Use of calculators is permitted.**
- **Write all answers in the space provided on the examination paper.**
- **Show your work where appropriate.**
- **Attempt all parts of the test before dwelling on any one part.**

NAME _____ Model Answers _____ STUDENT #: _____

PART	MARK
I.	/24
II.	/17
III.	/14
IV.	/19
V.	/11
VI.	/15
Total	**/100**

PART I. [24 marks]

A. [6 marks]

For each implication below, circle either "TRUE" or "FALSE" to indicate the validity of the implication. (The notation used here is the same as that used by Jones and Jones.)

Code C is *instantaneous* ==> C is a *prefix* code	**TRUE** FALSE
C is a *prefix* code ==> Code C is *uniquely decodable*	**TRUE** FALSE
C is *instantaneous* ==> $\sum_{i=1}^{q} r^{-l_i} \leq 1$	**TRUE** FALSE
$\sum_{i=1}^{q} r^{-l_i} = 1$ ==> $p_i = r^{-l_i}$ for all i	**TRUE** FALSE
X and Y independent ==> $H(X \mid Y) = H(X)$	**TRUE** FALSE
$H(X \mid Y) = 0$ ==> $H(X) \geq H(Y)$	TRUE **FALSE**

B. [4 marks]

(i). When coding for a *noiseless channel*, when treating a source alphabet of size n with probabilities p_1, ... p_n, what is the ideal average message length for a code? (Circle one.)

n $\quad$ $ceil(\log_2(n))$ $\quad$ ***[$H(p_1, \cdots, p_n)$]*** $\quad$ 10

(ii). When coding for a *noisy channel* and coding blocks of k bits with the intent to *correct* up to t single bit errors per codeword, what should the encoding length of each block be?

[$k+2t+1$] $\quad$ $k+t$ $\quad$ k $\quad$ $\log_2(k)1$

C. [6 marks]
For an input source with messages {a, b, c, d}, consider each of the six codes in the table below.

msg	code (i)	code(ii)	code (iii)	code (iv)	code (v)	code (vi)
a	0	1	0	000	0	00
b	01	10	1110	01	111	01
c	10	100	10	10	10	10
d	110	000	110	110	110	11

(i). For each code, by circling one of "AMB", "UD", "INST", indicate whether the code is *ambiguous*, *uniquely decodable*, or *instantaneous*, respectively.

code (i)	code(ii)	code (iii)	code (iv)	code (v)	code (vi)
AMB	AMB	AMB	AMB	AMB	AMB
UD	**UD**	UD	UD	UD	UD
INST	INST	**INST**	**INST**	**INST**	**INST**

(ii). For each code that is *instantaneous*, indicate whether or not there exists a set of probabilities for {a, b, c, d} such that the code is a *Huffman Code* for the set of messages. (Circle "YES" or "NO".)

code (i)	code(ii)	code (iii)	code (iv)	code (v)	code)vi)
YES	YES	YES	YES	**YES**	**YES**
NO	NO	**NO**	**NO**	NO	NO

D. [8 marks]

Consider the game of "Rock, Paper and Scissors" (abbreviated as R, P, S). In each round of this game, each of two players picks one object among R, P, and S. If both players pick the same object, the result is a "tie" with nobody winning. If the players pick different objects, then paper *covers* rock ($P > R$), rock *breaks* scissors ($R > S$), and scissors *cut* paper ($S > P$), so the winner in each case is as shown in the table below.

Case (i) Ch2: / Ch1:	R	P	S
R	NBDY	PL2	PL1
P	PL1	NBDY	PL2
S	PL2	PL1	NBDY

Define the random variables as follows:

$Ch1$ (= R or P or S) is the choice of Player 1
$Ch2$ (= R or P or S) is the choice of Player 2
W (= $PL1$ or $PL2$ or $NBDY$) is the winner as indicated in the table above

Assume that Player 1 chooses uniformly at random among R, P, and S (i.e., each on with probability $1/3$) in every round. Player 2, however, has two different behaviours in the two cases described below. For each of the two cases, fill in the entropy quantities in the table below.

(i). Assume in Case (i) that, like Player 1, Player 2 also chooses uniformly at random among R, P, and S in each round.

(ii). Assume in Case (ii) that Player 2 chooses "R" or "P" in each round, each with probability one-half, but *never* chooses "S".

Quantity	Case (i)	Case (ii)
$H(Ch1) =$	1.582	1.582
$H(Ch2) =$	1.582	1.000
$H(W) =$	1.582	1.582
$H(Ch1, Ch2) =$	3.164	2.582
$H(W, Ch1) =$	3.164	2.582
$H(W \mid Ch1) =$	1.582	1.000
$H(Ch1 \mid W) =$	1.582	1.000
$I(Ch1, W) =$	0	.582

HINT: All 16 correct answers are among these 6 values:

$\log_2(1) = 0$ $\log_2(2) = 1$ $\log_2(3) = 1.582$

$\log_2(3) - 1 = 0.582$ $\log_2(6) = 2.582$ $2 \times \log_2(3) = 3.164$

PART II. [17 marks]

A. [2 marks]
Relative Entropy can be used to measure the penalty of encoding based on the assumption that message probabilities follow distiribution q, when in fact they follow distribution p. For message set {a, b, c, d}, consider coding based on assuming a *uniform* distribution, $\vec{q}$ = (.25, .25, .25, .25), when the true distribution is $\vec{p}$ = (.50, .25, .125, .125). What is the differential entropy, $d(p \mid\mid q)$, which is the cost in bits per message due to misestimating the distribution? (Circle one.)

$$.5 \log_2(\frac{.50}{.25}) + .25 \log_2(\frac{.25}{.25}) + .25 \log_2(\frac{.125}{.25}) = .50 + 0.00 - .25 = .25$$

2.00	1.75	1.50	1.25	1.00	.75	.50	**.25**	0

B. [6 marks] Assume that a source generates messages from the set {a, b, c} with independent probabilities p_a = .8, p_b = .1, and p_c = .1.

(i). Specify a binary *Huffman Code* for the messages based on these probabilities.

a - 0 ; b - 10 ; c - 11

(ii). For the input message sequence, "c a a a c", give the bit sequence emitted using the Huffman code from part (i).

1100011

(iii). Consider using *Arithmetic Coding* to produce a binary encoding of the same source. Let "a" correspond to the interval [0.0, 0.8), "b" correspond to [0.8, 0.9), and "c" correspond to [0.9, 1.0). After the input message sequence, "c a a a c", what is the remaining subinterval of [0.0, 1.0) expressed in base 10? (That is, what is the interval in which the encoded value of *every* string that starts with "c a a a c ... " must lie?)

msg	-	c	a	a	a	c
lo	0.0	0.9	0.9	0.9	0.9	0.94608
hi	1.0	1.0	0.98	0.964	0.9512	0.9512

binary 1111001000 ... = .9453125
binary 1111010000 ... = .953125

(iv). Using arithmetic coding as in part (iii), after the first five source messages:
"c a a a c",
how many initial bits of the encoded string are known, and what are they?

Number of bits known: 7	Known bit(s): 1111001

C. [4 marks]
Message set {A, B, C} is to be coded with different variants of Huffman and Arithmetic Coding. *Initially,* the estimated probabilities are $p_A = .50$, and $p_B = p_C = .25$. Consider applying each of the four variants in the table below to the particular input string:

A B A C A B A C C C C C C C C C ... Note that this string initially conforms for the expected probabilities, but from the 9th message on, it deviates greatly from the intially assumed probabilities.

For each of the four algorithms, circle the *best estimate* of the average encoding length *per input message* as the number of "C"s at the end of the input string approaches infinity.

(non-adaptive) Huffman Coding	***2***	$\log_2(3)$	1.5	1.0	0
Adaptive Huffman Coding	2	$\log_2(3)$	1.5	***1.0***	0
(non-adaptive) Arithmetic Coding	***2***	$\log_2(3)$	1.5	1.0	0
Adaptive Huffman Coding	2	$\log_2(3)$	1.5	1.0	***0***

D. [5 marks]
Now consider using Lempel-Ziv=Welch coding for the message set {A, B, C}. Initially, the dictionary holds only the three entries, "A", "B", and "C". Consider processing the input sequence:

A A B B B C C C C A A A A A ...

In the first three iterations of the LZW algorithm, codes for "A", "A", and "B" will be transmitted, and additional dictionary entries "AA", "AB", and "BB" will be created.

For the string above, indicate what the next five strings to be added to the dictionary will be (i.e., the 7th through 11th entries).

7th: BBC	8th: CC	9th: CCC	10th: CA	11th: AAA

PART III. [14 marks]

A. [6 marks]

Given a very large body of English natural language text (e.g., a novel), consider compressing it by one of:

- Huffman coding based on letter frequencies (order 0 Markov chain)
- Huffman coding based on letter pair frequencies (order 1 Markov Chain)
- Huffman coding based on letter triple frequencies (order 2 Markov Chain)
- Prediction by Partial Match (PPM)

(i). In each row of the table below, enter digits 1, 2, 3, and 4 to indicate respectively the best, 2nd best, 3rd best and worst of the algorithms with respect to each criterion in the first column.

Criterion	0th order	1st order	2nd order	PPM
best compression achieved	4	3	2	1
least computational cost	1	2	3	4
least space required	1	2	3	4

(ii). If these algorithms were applied not to a body of English text, but rather to a sequence of letters in which each letter is selected according to a fixed probability distribution *independently* of all previous letters, describe *briefly* how the rankings in the table above would change (if at all).

If successive characters are independent, then all four techniques will lead to the same degrees of compression (since dependence can't be exploited by the Markov models. The relative amounts of computation and space consumed by the four techniques will remain as before.

B. [8 marks]
For the source alphabet { a , b , c }, two different *first-order Markov models* are shown below, along with their corresponding *state-transition probabilities*. For each of the two, enter the values for the following quantities into the table below:

(i). What are the equilibrium probabilities of each state, $\{T_a, T_b, T_c\}$?

(ii). What is the *entropy*, $H(X_n)$, of the state?

(iii). What is the *entropy rate*, $H(X_n \mid X_{n-1})$?

Model I Model II

Transition Matrices:

	a	b	c
a	½	½	0
b	½	0	½
c	0	½	½

	a	b	c
a	¼	¼	½
b	½	0	½
c	½	¼	¼

	Model I	Model II
(i) $\{T_a, T_b, T_c\}$	{ 1/3 , 1/3 , 1/3 }	{ .4 , .2 , .4 }
(ii) $H(X_n)$	$H(1/3, 1/3, 1/3)$	$H(.4, .2, .4)$
(iii) $H(X_n \mid X_{n-1})$	1.0	1.4

PART IV. [19 marks]

A. [8 marks]

Below is a picture of an *Asymmetric Binary Channel*. In this channel, "0"s are always received correctly, but "1"s are received correctly just 80% of the time; the other 20% of the time, they are transformed into "0"s. Assume that the input probabilities are $p_0 = 0.6$ and $p_1 = 0.4$, or

$$A_i = \begin{bmatrix} 0.6 & 0.4 \end{bmatrix}.$$

In the spaces below, fill in the entries of the various matrices to reflect the characteristics of this channel.

Channel Matrix:

$$P_{ij} = \begin{bmatrix} 1.0 & 0.0 \\ 0.2 & .8 \end{bmatrix}$$

Joint Probability Matrix:

$$R_{ij} = \begin{bmatrix} 0.6 & 0.0 \\ 0.08 & .32 \end{bmatrix}$$

Backward Probabilities:

$$Q_{ij} = \begin{bmatrix} 0.882 & 0.0 \\ 0.118 & 1.0 \end{bmatrix}$$

Output Probabilities:

$$B_i = \begin{bmatrix} 0.68 & 0.32 \end{bmatrix}$$

B. [3 marks]

Compare the following three families of codes with repsect to their asymptotic behaviour as the length of the codewords approaches infinity ($n \to \infty$):

- *Repetition codes* with a single message bit, Repetition $[n, 1]$
- *Parity codes* with a single check digit, Parity $[n, n-1]$
- *Hamming codes* with c check digits, Hamming $[n, n-c]$

In the table below, for each of the three families, circle either zero or one to indicate the asymptotic *rate of transmission* and the asymptotic *probability of error* in decoding a codeword as $n \to \infty$.

As $n \to \infty$:	Repetition $[n, 1]$		Parity $[n, n-1]$		Hamming $[n, n-c]$	
Rate of Transmission	**0**	1	0	**1**	0	**1**
Probability of Error	**0**	1	0	**1**	0	**1**

C. [8 marks]

For a binary source, with messages {0, 1}, consider using a *5-repetition code (n = 5) consisting of codewords {00000, 11111} to transmit the messages over a noisy channel, which causes each bit to be received correctly with probability P, and incorrectly with probability* 1–*P*.

(i). With *maximum likelihood decoding*, what is the largest number of errors in the transmission of a codeword that will be *corrected*?

2

(ii). Give an expression in terms of P for the probability that the message will be received *correctly* with 5-repetition coding.

$$P^5 + 5P^4 \times (1 - P) + 10P^3 \times (1 - p)^2$$

(iii). Let the probabilities of "0" and "1" in the input stream be p_0 and p_1, respectively. Under what condition (stated in terms of p_0, p_1 and P) would *ideal observer decoding* cause the sequence "11100" to be interpretted as "0" rather than "1" in order to minimize the probability of error?

$$p_0 (1 - P) \geq p_1 P$$

(iv). Under what condition (stated in terms of p_0, p_1 and P) would *ideal observer decoding* cause *every* sequence of 5 bits received (even "11111") to be interpretted as "0" in order to minimize the probability of error?

$$p_0 (1 - P)^5 \geq p_1 P^5$$

PART V. [11 marks]

A. [4 marks]

(i). For the continuous uniform distribution from 0 to 4, Uniform[0, 4], ...
What is its *differential entropy*?

$$\log_2(4) = 2$$

How many bits are required at minimum to represent samples from this distribution to the nearest 1/16th?

$$2 + 4 = 6$$

(ii). For the continuous uniform distribution from 0 to .25, Uniform[0, .25], ...
What is its *differential entropy*?

$$\log_2(.25) = -2$$

How many bits are required at minimum to represent samples from this distribution to the nearest 1/16th?

$$-2 + 4 = 2$$

B. [2 marks]

When real numbers from a continuous distribution are represented in a computer in standard floating point representation:

$r = f \times b^e$, where b is the base, e is the exponent, and f is the fraction,

which type of *Quantization* is being employed? (Circle one.)

Uniform Quantization	**Non-uniform Quantization**	Adaptive Quantization

C. [5 marks]

(i). *Scalar Quantization* with $L = 2$ representatives has been performed in various ways on the continuous distribution:

$$f(x) = \begin{cases} 1/2 & 0 \le x \le 1 \\ (3-x)/4 & 1 \le x \le 3 \\ 0 & \textit{otherwise} \end{cases}$$

The table below indicates the boundaries $\{ b_0, b_1, b_2 \}$ and the representatives $\{ r_1, r_2 \}$ produced by each method. In the column labelled "Method", enter one of: (a) Uniform Quantization, (b) Equal Interval Weight Quantization, and (c) Optimal Quantization, to indicate the method of quantization used to produce that row.

$b_0 =$	$r_1 =$	$b_1 =$	$r_2 =$	$b_2 =$	Method
0	.50	1.00	1.59	3.00	(b)
0	.55	1.10	1.66	3.00	(c)
0	.72	1.50	1.94	3.00	(a)

(ii). For the same distribution as in part (i), *Vector Quantization* with a vector length of $k = 2$ and with $L = 4$ representatives per *pair* of inputs has been performed. The results of four different methods are shown in the four diagrams below. Use your *judgement* and *intuition* to determine how well you believe each of the four methods will do with respect to minimizing the resulting distortion. Indicate your ordering by entering the identifiers W, X, Y, Z in the table below.

W {(.50,.50),(.50,2.0),(2.0,.50),(2.0,2.0)}

X {(.55,.55),(.55,1.65),(1.625,.55),(1.625,1.625)}

Y: {(.60,.60),(.5,1.7),(1.7,.5),(1.5,1.5)}

Z: {(.75,.75),(.75,2.25),(2.25,.75),(2.25,2.25)}

Best	2nd best	3rd best	Worst
Y or X	X or Y	W	Z

PART VI. [15 marks]

A. [8 marks]

Consider applying a simplified *wavelet* transform (as illustrated in lecture) to an input sequence, $v_1, v_2, \cdots, v_n$, and then transmitting some subset of the transform coefficients to a receiver for the purpose of reconstructing the original sequence as $r_1, r_2, \cdots, r_n$. Let *distortion* be measured by: $D = \frac{1}{n}\sum_{i=1}^{n} |v_i - r_i|$ and let the input sequence to which the transform is applied be:

16 12 2 10

$a_{12} = 14$, $d_{12} = -2$ $\quad$ $a_{34} = 6$, $d_{34} = +4$

$a_{14} = 10$, $d_{14} = -4$

(i). What are the four resulting transform coefficients?

$a_{14} = 10$ $\quad$ $d_{14} = -4$ $\quad$ $d_{12} = -2$ $\quad$ $d_{34} = +4$

(ii). Consider transmitting a subset of the transform coefficients to a receiver, and using them reconstruct an estimate of the original signal. For each of the three subsets shown in the first column of the table below, fill in both the *reconstructed signal* and the resulting *distortion*.

Subset	r_1, r_2, r_3, r_4	Distortion, D
a_{14} only	10 10 10 10	4
a_{14} and d_{14}	14 14 6 6	3
a_{14}, d_{14}, d_{12}, and d_{34}	16 12 2 10	0

B. [3 marks]
Consider an audio input source composed of frequncies in the range 20 Hz to 20 kHZ (i.e., from 20 cycles per second to 20,000 cycles per second). The source is sampled at rate R samples per second, and the sampled values are quantized at L levels.

(i). If the sampling rate is R = 10,000 samples per second, what is the upper limit on the range of frequencies (expressed in cycles per second) that will be recoverable with *no* distortion due to sampling (ignoring the distortion due to quantization)? (Circle one.)

5,000	10,000	20,000	40,000	80,000

(ii). How high would the sampling rate have to be (expressed in samples per second) in order to assure that there would be no distortion due to sampling over the full range of frequencies from 20 Hz to 20 kHz? (Circle one.)

5,000	10,000	20,000	**40,000**	80,000

(iii). Assuming that the sampling rate is sufficiently high that distortion due to sampling is avoided, how many levels L must be used in quantization in order to assure that there will be no distortion due to quantization? Explain your answer briefly.

No matter how large L is made, there will always be some distortion due to the quantization.

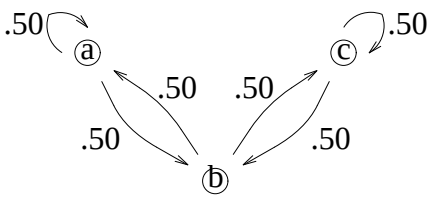
C. [4 marks]
Consider compressing 1024 × 1024 pixel gray-scale images of natural scenes by each of the following methods:

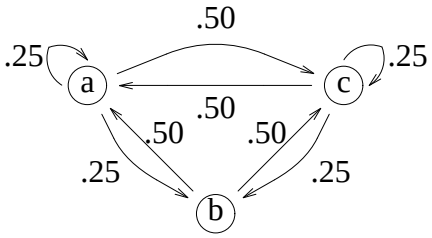
- HUFF : Huffman encode the value of each successive pixel independently
- MRKV : Form a 1st order Markov chain representation of the values in successive pixels across a row, and Huffman encode based on the transition probabilities of the Markov chain
- BLCK : Apply a transform to each 8 by 8 block of pixels, and encode for each block the values of the coefficients of the transform

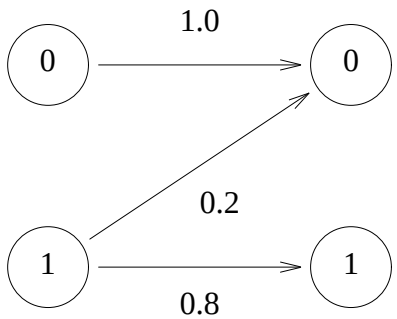
Comment briefly on which of these will lead to the *least compression* and which will lead to the *greatest compression*, and indicate why.

HUFF is the worst because it fails to exploit dependency in pixel values either horizontally or vertically. BLCK is the best because it does exploit these dependencies. MRKV is in between because it exploits horizontal but not vertical dependencies.

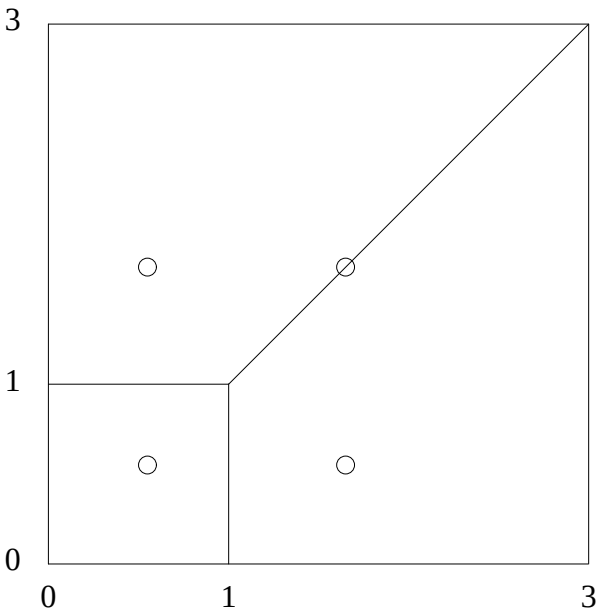
Case (ii) Ch1:	Ch2: R	P
R	NBDY	PL2
P	PL1	NBDY
S	PL2	PL1



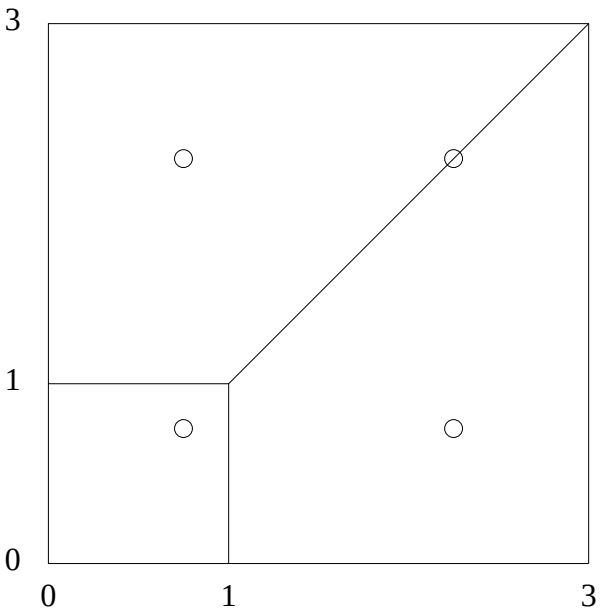




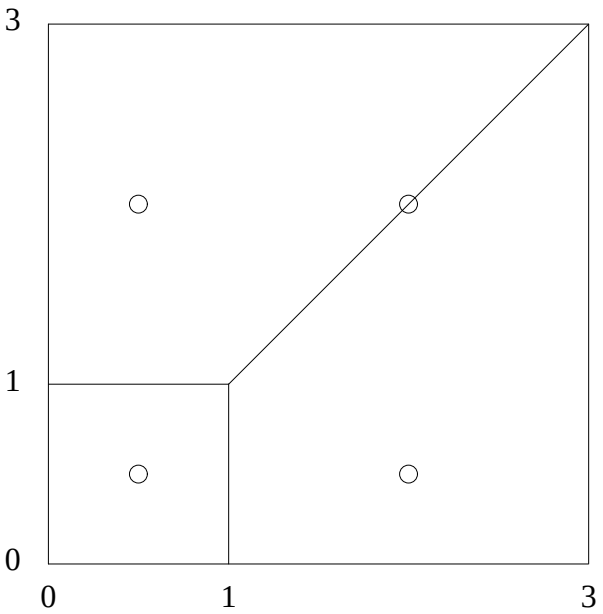
X



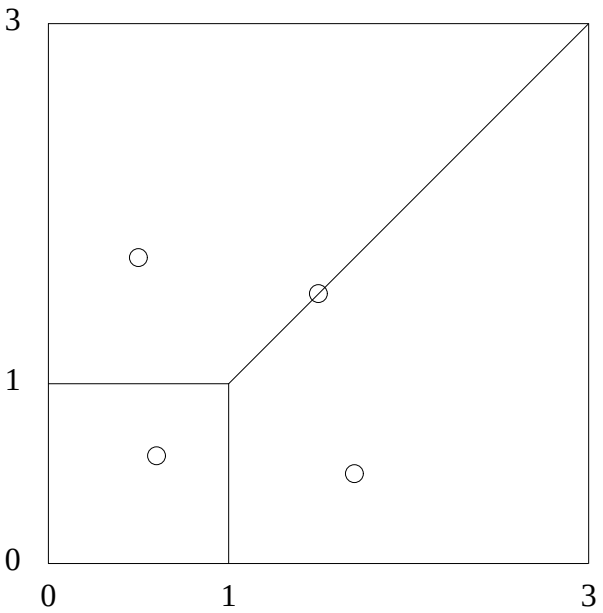
Z

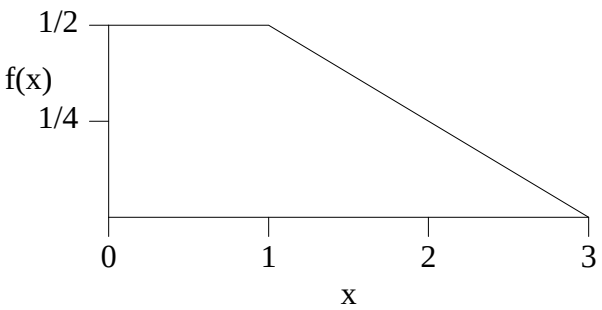


W



Y





Page 1 of 14

Page 2 of 14

Page 3 of 14

Page 4 of 14

Page 5 of 14

Page 6 of 14

Page 7 of 14

Page 8 of 14

Page 9 of 14

Page 10 of 14

Page 11 of 14

Page 12 of 14

Page 13 of 14

Page 14 of 14

TOTAL PAGES = 14

TOTAL MARKS = 100