Solutions for 2015 EEE6222

Q1(a)

(i) The Nyquist minimum bandwidth for detecting r symbols per second without ISI is r/2 Hz. Therefore for a system with bandwidth 4kHz the maximum symbol rate without ISI is $r_{\max}$ = 2*4000 = 8000 symbols/second.
(1 mark)
As the symbol period T=1/r=1/8000 second, the ideal Nyquist pulse function in this case is given by $h(t) = \frac{\sin(\pi t/T)}{\pi t/T} \approx \frac{\sin(25120t)}{25120t}$
(2 marks)

(ii) Channel capacity is given by

S/N=10^(20/10)=100;
C = B*log2(1 + S/N) = 6.658 x 4000 =26632 bits/second
(2 marks)

To achieve a data rate near to this a practical system would need to use M-ary signalling (to increase the number of bits/symbol and hence the bandwidth efficiency) and also use some form of coding scheme to minimise errors.
(1 mark)

(iii) The entropy of the sampled signal is H=1/2+1/2+3/8+1/4+1/4=15/8 bits/sample. Given the channel capacity C=26632 bits/s, the maximum number of samples transmitted without error is R=C/H=14204 samples/s.
(2 marks)

Q1(b)

E_b/N_0 is related to the system signal to noise ratio through the following identities for binary systems (i.e. systems that send 1 bit per symbol):

$$\frac{E_b}{N_0} = \frac{ST}{N_0} = \frac{S}{RN_0} = \frac{S}{N_o B}\left(\frac{B}{R}\right)$$

Where B = bandwidth
S = average signal power
T = symbol (bit) time period
R = bit rate (data rate)

For the case where transmission bit rate is equal to the channel capacity, R=C, we have

$$\frac{E_b}{N_0} = \frac{S}{N_o B}\left(\frac{B}{C}\right)$$

Using

$$\frac{C}{B} = \log_2(1+\frac{S}{N_o B})$$

i.e.

$$\frac{C}{B} = \log_2\left(1+\frac{E_b}{N_o}\left(\frac{C}{B}\right)\right)$$

let

$$x = \frac{E_b}{N_o}\left(\frac{C}{B}\right)$$

then

$$\frac{C}{B} = x\log_2(1+x)^{\frac{1}{x}}$$

i.e.

$$1 = \frac{E_b}{N_o}\log_2(1+x)^{\frac{1}{x}}$$

(3 marks)

In the limit, as $\frac{C}{B} \to 0$ and using

$$\lim_{x\to 0}(1+x)^{1/x} = e$$

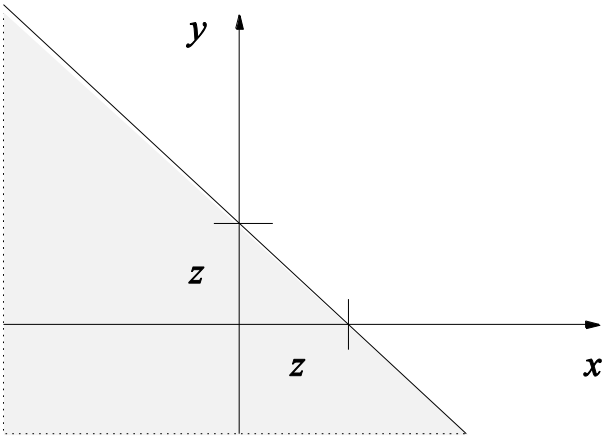
(2 marks)
we get

$$\frac{E_b}{N_o} = \frac{1}{\log_2 e} = 0.693 = -1.59dB$$

(1 mark)

Q1(c)

When x and y are random variables the probability of their sum can be found from a consideration of the following figure:



(1 mark)

$$P(x+y \le z) = P\{(x,y)\ in\ shaded\ area\}$$
$$= \int_{-\infty}^{\infty}\int_{-\infty}^{z-x} f_{X,Y}(x,y)\,dxdy$$
$$f_Z(z) = \frac{d}{dz}\int_{-\infty}^{\infty}\int_{-\infty}^{z-x} f_{X,Y}(x,y)\,dxdy$$
$$= \int_{-\infty}^{\infty} f_{X,Y}(x,z-x)dx$$

(3 marks)

For statistically independent variables, we have

$$f_Z(z) = \int_{-\infty}^{\infty} f_X(x) fY(z-x)dx$$
$$= \int_{-\infty}^{\infty} f_X(z-y) fY(y)dy$$

(2 marks)

Q2(a)

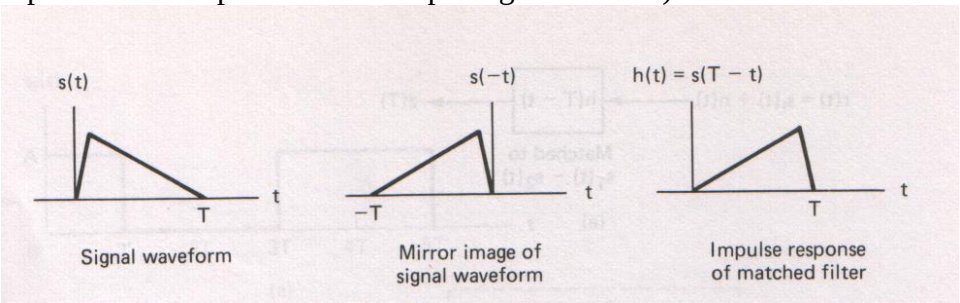
A matched filter attempts to maximise the signal to noise ratio at the input to the detector. This can be achieved using a filter design that is *matched* to the specific transmitted signal waveform.
(1 mark)

It can be shown that for a symbol with waveform, $s(t),\quad 0 \le t \le T$, the filter that ensures maximum signal to noise ratio (SNR) at its output at time *T* will have an impulse response given by

$$h(t) = \begin{cases} ks(T-t), & 0 \le t \le T \\ 0, & \text{elsewhere} \end{cases}$$

(1 mark)

This is easily seen to be the mirror image of the original signal delayed by *T*. This delay is necessary to make the filter realisable (without the delay, the filter would have to produce an output before the input signal arrived!)



(1 mark)

The output from the filter for a received signal r(t) is given by

$$y(t) = r(t) * h(t) = \int_0^t r(\tau)h(t-\tau)\,d\tau$$

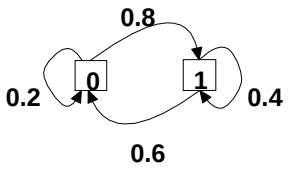
using a matched filter with k = 1, we see that at time T the out put is

$$y(T) = \int_0^T r(\tau)s(\tau)\,d\tau$$

This expression is known as the correlation of r(t) and s(t) and gives a measure of the similarity between the two signals. If r(t) and s(t) are the same, then the correlation value will be high.

Thus we see that the output from the matched filter will be high for a received signal that matches the signal used to design the filter, other signals will produce a low value.
(2 marks)

Q2(b)



(i) Let the source be referred to as X then the Entropy is given by:
H(X) = P(0)*H(X|0) + P(1)*H(X|1).

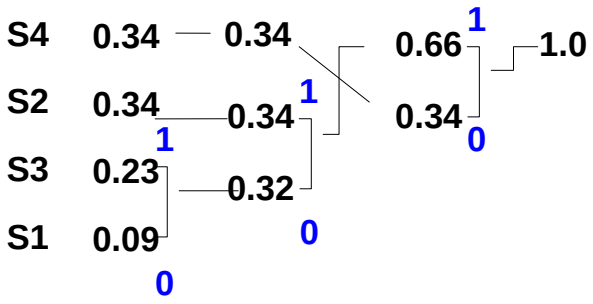
We begin by solving for the *a priori* probabilities. This can be done using the fact that:
P(1) + P(0) = 1
Substituting this into
P(0) = P(0)P(0|0) + P(1)P(0|1)
and using the conditional probabilities indicated in the Markov model [P(1|1) = 0.4, etc] we get P(0) = 3/7 and therefore P(1) =4/7 .
(2 marks)

The conditional entropies H(X|0) and H(X|1) are then computed:
H(X|0) = P(0|0)*log2(1 / P(0|0)) + P(1|0)*log2(1 / P(1|0)) = 0.7219
H(X|1) = P(1|1)*log2(1 / P(1|1)) + P(0|1)*log2(1 / P(0|1)) = 0.9710
(2 marks)

So finally,
H(X) = P(0)*H(X|0) + P(1)*H(X|1) = 0.7219*3/7+0.9710*4/7 = 0.8642 bits / symbol
(1 mark)

(ii) To determine a unique, prefix-free coding scheme for this source close to its entropy, we can use extension code by using groups of two symbols as the basis and then apply Huffman coding to this new set of symbols. The new set of symbols are S1=00, S2=01, S3=11, and S4=10. Their associated probabilities are respectively:
p(00)=p(0)*p(0|0)=0.2*3/7= 6/70=0.086
p(01)=p(1)*p(0|1)=0.6*4/7=24/70=0.343
p(11)=p(1)*p(1|1)=0.4*4/7=16/70=0.228
p(10)=p(0)*p(1|0)=0.8*3/7=24/70=0.343

(2 marks)
Apply Huffman coding to these four symbols we have



So, the results are:
S1: 100
S2: 11
S3: 101
S4: 0
(2 marks)
[Note: variations are possible depending on the exact algorithm used. However, to be acceptable, the code must be uniquely decodable, prefix free and efficient]

(iii)
The average code length is given by:

$$\bar{n} = \sum_{i=1}^{4} n_i P_i$$

where n_i is the bit length of the codeword for symbol S_i, and P_i is the probability for S_i.
Average length $\bar{n}$ = 0.09*3+0.34*2+0.23*3+0.34*1=1.98 bits / symbol
Back to the original symbol (binary symbol), it is 1.98/2=0.99bits/symbol. Then the coding efficiency is

$$\frac{0.8642}{\bar{n}} = 0.8729$$

(1 mark)

Q2(c)

(i)

Message	Parity	Codeword
000	0	0 000
100	1	1 100
010	1	1 010
110	0	0 110
001	1	1 001
101	0	0 101
011	0	0 011
111	1	1 111

(1 mark)

(ii) The code is capable of detecting all single- and triple-error patterns. (1 mark)

(ii) The probability of an undetected error is equal to the probability that two or four errors occur anywhere in a codeword.
(1 mark)

$$\begin{aligned} P_{nd} &= \binom{4}{2} p^2 (1-p)^2 + \binom{4}{4} p^4 \\ &= 6p^2(1-p)^2 + p^4 \\ &\approx 2.4 \times 10^{-5} \end{aligned}$$

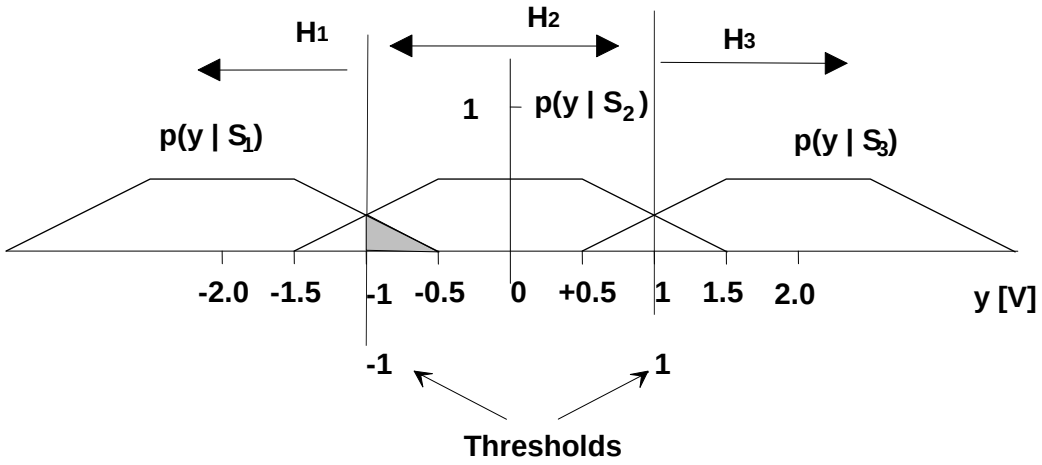
(2 marks)

Q3(a)

(i)For a maximum likelihood detector, we choose the hypothesis that has the highest likelihood value for the observed signal when compared with all possible hypotheses:

$$\hat{H}_{ML} = \arg\max_i \{p(y \mid S_i)\}$$

Considering a graphical representation of the situation:



(2 marks)
It is clear that the decision thresholds are at –1V and 1V. The full decision rule is as follows:
Choose H1 if y < -1,
Choose H2 if -1< y < 1, and
Choose H3 if y > 1
(1 mark)

(ii) The probability of error is calculated by considering all the different possible ways of making an error:
Pe = P(H1|S2)P(S2) + P(H1|S3)P(S3) + P(H2|S1)P(S1) + P(H2|S3)P(S3) + P(H3|S1)P(S1) + P(H3|S2)P(S2)
(2 marks)
From the diagram, it is clear that P(H1|S3) and P(H3|S1) are both zero.

P(H2|S1) is the shaded area under the graph of p(y|S1) so

$$P(H_2 \mid S_1) = \int_{-1}^{1} p(y \mid S_1)dy = 0.5 \times (\textit{triangle base}) \times (\textit{triangle height})$$

$$= 0.5 \times 0.5 \times 0.25$$

$$= 0.0625$$

(2 marks)

Also, because of symmetry, P(H1|S2) = P(H2|S3) = P(H3|S2) = P(H2|S1) so Pe = 4* P(H2|S1) * P(S1) = 0.0833

(1 mark)

(iii) If P(S2) → 1, then the decision thresholds would move outwards towards –1.5 and 1.5. In the limit the MAP error probability would tend to zero (If P(S2) = 1 then the receiver could always just choose H2 and would never make an error.)

(2 marks)

Q3(b)

(i) 0.4*0.5=0.2 (1 mark)

(ii) P (E_1U E_2) = P (E_1) + P (E_2) - P ($E_1 \cap E_2$)=0.4+0.5-0.2=0.7. (2 marks)

(iii) 1- P (E_1U E_2)=1-0.7=0.3. (1 mark)

Q3(c)

(i) The minimum Hamming distance between the codes is Dmin = 3. Therefore, Dmin - 1 = 2 bit errors can be detected per code word. [Note that some three bit errors could also be detected because the Hamming distance between some of the codes is greater than 3]

(2 marks)

(ii) Forward error correction is possible for 1 bit errors since a 1 bit error will result in a code that is closer to the original codeword than to any other valid codeword. (Dmin - 1)/2 = 1)

(2 marks)

(iii) For four messages we would need 2 bits / message. The linear block code in question uses n = 6 bits / codeword. Therefore the code rate is k/n = 2/6 = 33%.

(2 marks)

Q4(a)

When receiving digital signals it is necessary to synchronise the receiver with the incoming signal. There are three levels of synchronisation.

1. Synchronising the Carrier.

We need to lock the receiver to the incoming carrier frequency. We can use a Phase-Lock-Loop (PLL). Communication systems can operate without carrier synchronisation by using non-coherent detection, but the probability of error is then increased in most cases.

(1 mark)

2. Symbol synchronisation.
This is a higher level of synchronisation where the receiver tries to detect each individual symbol (i.e. each bit in a binary system). Bit synchronisation can be achieved using an Early/Late gate synchroniser.
(1 mark)
3. Frame synchronisation.
This is the highest level of synchronisation, where the aim is to identify blocks of data (frames) within the data stream. One approach would be to put a special voltage waveform in the signal (similar to having a full stop in a sentence) but this requires special circuitry. A better idea is to insert a code marker at the beginning of a frame and try and find this at the receiver. Barker code and Willard Code are two examples of such a code marker.
(1 mark)

Q4(b)

1. Is it DC Free? DC component is a wasted energy since many components in a communications system, e.g. transformers or filters, don't pass the dc component.
2. Does is provide for easy synchronisation? In some waveforms there is a level transition at the middle of bit interval. This transition is used at the receiver to achieve synchronisation with the transmitter.
3. Is it immune to signal inversion? If a positive voltage is used to represent one state and negative for the other, then if the signal pass through inverters, then the entire data will be inverted and every bit will be in error.
4. How well does it works in the presence of noise? Polar waveforms have a better SNR compare with uni-polar waveforms.
5. Does it facilitate error detection? The transition at the middle of the bit interval is used for error detection as well as synchronisation. Absence of the transition means there is an error.

(one mark for each point)

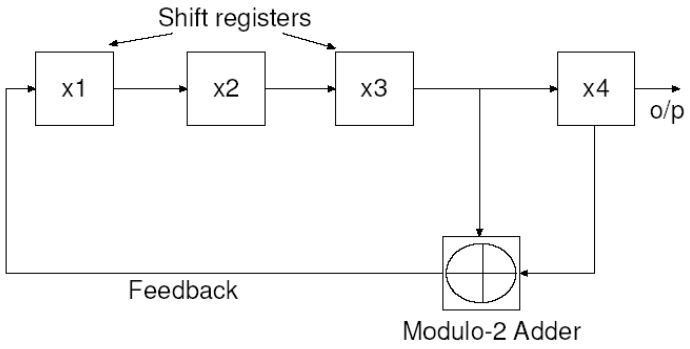
Q4(c)

(i) There are three properties:

(a) Balance property. We want roughly the same number of 0s as we have 1s in the sequence.
(b) Run property. A run sequence is a sequence of consecutive bits with the same value. We want
    a. 1/2 of run sequences to be of length 1
    b. 1/4 of run sequences to be of length 2
    c. 1/8 of run sequences to be of length 3…
(c) Correlation property. We would like the PN sequence to have very low correlation with shifted copies of itself (as with white noise). This also helps the receiver synchronise correctly.

(1 mark for each point)

(ii) PN sequences for spread spectrum systems can be produced using circuits consisting of shift registers with feedback. An example is given below.



(3 marks)

Q4(d)

i) The full communication process is as follows: (e = 3, d = 7, N = 33)

1. Alice publishes *N* and e as the public key for everyone to see, and keep d secret as the private key.
2. Before Bob sends the message *M* = 9 to Alice, he looks up *e* and *N* and calculates $C = M^e \bmod N = 3$. This is transmitted to Alice on a non-secure channel. Although other people can read this message, they would not be able to determine *M*, since a one-way function has been used to encrypt the message.
3. To decipher the message Alice uses the value only she knows, i.e. d=7. To get the real message Alice calculates $M = C^d \bmod N = 3^7 \bmod 33 = 9$.

(4 marks)

ii) The reason why Alice can decipher the message correctly is that the three figures (e = 3, d = 7, N = 33) are chosen in advance to satisfy the following equation

$$x = x^{ed} \bmod N.$$

Note C is obtained by $C = M^e \bmod N$. When Alice calculate $C^d \bmod N$, it is equivalent to calculate $(M^e \bmod N)^d \bmod N$. According to the following identity $p^q \bmod r = (p \bmod r)^q \bmod r$, we have

$(M^e \bmod N)^d \bmod N = (M^e)^d \bmod N = M^{ed} \bmod N = M$

(2 marks)

Q5(a)

The total number of permutations is 4*3*2*1=24.

We can list all of them:

a1,a2,a3,a4; a1,a2,a4,a3; a1,a3,a2,a4; a1,a3,a4,a2; a1,a4,a2,a3; a1,a4,a3,a2

a2,a1,a3,a4; a2,a1,a4,a3; a2,a3,a1,a4; a2,a3,a4,a1; a2,a4,a1,a3; a2,a4,a3,a1

a3,a1,a2,a4; a3,a1,a4,a2; a3,a2,a1,a4; a3,a2,a4,a1; a3,a4,a1,a2; a3,a4,a2,a1

a4,a1,a2,a3; a4,a1,a3,a2; a4,a2,a1,a3; a4,a2,a3,a1; a4,a3,a1,a2; a4,a3,a2,a1

(2 marks)

$P(E_1)$=3*2*1/24=1/4;

$P(E_3)=3*2*1/24=1/4$;
(1 mark)
For $P(E_1 \cup E_3)=10/24=5/12$
(1 mark)
For $P(E_1 \cap E_3)=2/24=1/12$
(1 mark)
$P(E_1) + P(E_3) - P(E_1 \cap E_3)=1/2-1/12=5/12= P(E_1 \cup E_3)$.
Verified.
(1 mark)

Q5(b)

The question is to ask the probability of having a 1 at the other side, given the condition that the first side is a 1. We use E1 to represent the event that the first side is 1 and E2 to represent the event that the second side is a 1.
Then $P(E1 \cap E2)=1/3$ (i.e. the probability of taking the card with both sides being 1)
(1 mark)
$P(E1)=1/3+1/3*(1/2)=1/2$ (i.e. the sum of the probability of taking the first card and the probability of taking the third card plus taking the side with 1)
(1 mark)
So $P(E2|E1)= P(E1 \cap E2)/P(E1)=2/3$
(2 marks)

Q5(c)

(i)
The error rate $\varepsilon=0.01$.
Let S_1 be the input with E1=0 and E2=1, and S_2 the output with F1=0 and F2=1. Then with equal probability at the input and also symmetric channel, we have
$P(E1)=P(E2)=P(F1)=P(F2)=0.5$.
(1 mark)
$P(E2 \cap F2)=P(F2|E2)P(E2)=0.5(1-\varepsilon)$
(1 mark)
$I(E2,F2)=\log_2(P(F2,E2)/P(E2)/P(F2))= \log_2 2(1-\varepsilon)=0.986$ bits.
(1 mark)

(ii)
$P(E2 \cap F1)=P(F1|E2)P(E2)=0.5\varepsilon$
(1 mark)
$I(E2,F1)=\log_2(P(F1,E2)/P(E2)/P(F1))= \log_2 2(\varepsilon)=-5.64$ bits.
(1 mark)

(iii)
The mutual information between the two systems is
$H(S_2)=1$ bit

(1 mark)

$$H(S_2|S_1)=\frac{-1}{2}(1-\varepsilon)\log(1-\varepsilon)+\frac{-1}{2}\varepsilon\log\varepsilon+$$
$$\frac{-1}{2}\varepsilon\log\varepsilon+\frac{-1}{2}(1-\varepsilon)\log(1-\varepsilon)$$
$$=-(1-\varepsilon)\log(1-\varepsilon)-\varepsilon\log\varepsilon$$

(2 mark)

$I(S_1,S_2)=H(S_2)-H(S_2|S_1)=1-(-(1-\varepsilon)\log_2(1-\varepsilon)-\varepsilon\log_2\varepsilon)=0.92$ bits.

(2 mark)

Q6(a)

Suppose there are M antennas at the transmitter and also M antennas at the receiver. Each of the transmit antennas sends a symbol at time n and together there are M symbols sent. We denote them by a vector X. At the receiver side, we also receive M symbols, denoted by a vector Y:

$$X=\begin{bmatrix} x_0 \\ \vdots \\ x_{M-1} \end{bmatrix} \qquad Y=\begin{bmatrix} y_0 \\ \vdots \\ y_{M-1} \end{bmatrix}$$

They have the following relationship: $Y=HX+N$

(2 marks)

Where H is the channel response matrix and N is the noise vector.

$$\begin{bmatrix} y_0 \\ \vdots \\ y_{M-1} \end{bmatrix}=\begin{bmatrix} h_{0,0} & \cdots & h_{0,M-1} \\ \vdots & \ddots & \vdots \\ h_{M-1,0} & \cdots & h_{M-1,M-1} \end{bmatrix}\begin{bmatrix} x_0 \\ \vdots \\ x_{M-1} \end{bmatrix}+N$$

(1 mark)

If we know the channel response matrix H (by some channel estimation methods) and it has full rank (which is usually true in strong multipath environments with antennas spaced sufficiently far away from each other), then given the received symbols Y, we can recover the original symbols X by matrix inversion or the maximum likelihood method, even if some of the values in H are zeros.

(2 marks)

Note for a SISO (single input and single output) system, if the channel response is zero, we will not be able to recover the original signals.

(1 mark)

Q6(b)

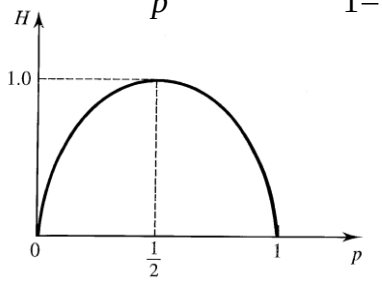
(i) The **entropy** of a source S, called H(S), is the average of the self-information, i.e.

$$H(S)=\mathrm{E}(I)=-\sum_{k=1}^{n} p_k\log p_k$$

where p_k is the probability of the event E_k of this source and there are in total *n* different events. It is a measure of the source's randomness. The more random a source is, the higher its entropy.
(2 marks)

(ii) The highest entropy occurs when the events (or symbols) have equal probabilities. As an example, consider a two symbol (binary) source where *p* is the probability of transmitting a '1' we have:

$$H = p\log_2\frac{1}{p} + (1-p)\log_2\frac{1}{1-p}$$



The maximum entropy value 1.0 is achieved when we have p=0.5.
(4 marks)

(iii) To prove this condition, we need to use the following theorem:
For x>0, lnx≤x−1, with equality only when x=1.
(2 marks)

Assume firstly that no p_k is zero. Then,

$$\sum_{k=1}^{n} p_k \ln\frac{1}{np_k} \le \sum_{k=1}^{n} p_k\left(\frac{1}{np_k} - 1\right)$$

$$=\sum_{k=1}^{n}\left(\frac{1}{n} - p_k\right) = 1 - 1 = 0$$

$$-\sum_{k=1}^{n} p_k \ln p_k \le \sum_{k=1}^{n} p_k \ln n = \ln n$$

Hence,
ln can be converted to log by multiplying by a suitable positive constant. Since multiplication by a positive constant does not invalidate an inequality, we have

$$-\sum_{k=1}^{n} p_k \log p_k \le \log n \quad \text{or} \quad H(S) \le \log n$$

Since every p_k is positive, equality will arise only if it does for every term in the First Result. So we have $1/(np_k)=1$ for all k. The theorem is therefore proved for non-zero probabilities.
(4 marks)

If a probability, say p_j , is zero, $p_j \ln p_j = 0$, and

$$p_j \ln\frac{1}{np_j} < \frac{1}{n} - p_j$$

Thus the first result continues to hold and the upper bound on H(S) then follows. The above inequality prevents equality in the theorem. The proof is finished.
(2 marks)