3 problems, 85 points
165 minutes
1 sheet (2 pages) of notes allowed.

Good Luck!

PLEASE WRITE YOUR NAME ON EACH SHEET OF YOUR ANSWERS.

PLEASE WRITE THE SOLUTION OF EACH PROBLEM ON A SEPARATE SHEET.
Problem 1. (25 points) Suppose $X, Y$ and $Z$ are random variables.

(a) (5 pts) Show that $H(X) + H(Y) + H(Z) \geq \frac{1}{2} [H(XY) + H(YZ) + H(ZX)].$

(b) (5 pts) Show that $H(XY) + H(YZ) \geq H(XYZ) + H(Y).$

(c) (5 pts) Show that


(d) (5 pts) Show that $H(XY) + H(YZ) + H(ZX) \geq 2H(XYZ).$

(e) (5 pts) Suppose $n$ points in three dimensions are arranged so that their projections to the $xy$, $yz$ and $zx$ planes give $n_{xy}$, $n_{yz}$ and $n_{zx}$ points. Clearly $n_{xy} \leq n$, $n_{yz} \leq n$, $n_{zx} \leq n$. Use part (d) show that

$$n_{xy} n_{yz} n_{zx} \geq n^2.$$
Problem 2. (30 points) Consider the distribution $Q$ on the positive integers $\{1, 2, \ldots\}$ with

$$Q(u) = (1 - 2p)p^{\left\lfloor \log_2(u) \right\rfloor}, \quad u = 1, 2, \ldots,$$

i.e.,

- $Q(1) = (1 - 2p)$,
- $Q(2) = Q(3) = (1 - 2p)p$,
- $Q(4) = Q(5) = Q(6) = Q(7) = (1 - 2p)p^2$,
- $\ldots$,
- $Q(2^j) = \ldots = Q(2^{j+1} - 1) = (1 - 2p)p^j$, \quad ($j = 0, 1, \ldots$).

Suppose the random variable $V$ has distribution $Q$.

(a) (5 pts) Find $H(V)$. [Hint: $\sum_{j=0}^{\infty} x^j = 1/(1 - x)$, $\sum_{j=1}^{\infty} jx^j = x/(1 - x)^2$.]

(b) (5 pts) Find $L(V) := E[\lfloor \log_2 V \rfloor]$.

(c) (5 pts) Show that

$$H(V) = L(V) + L(V) \log_2(1 + 1/L(V)) + \log_2(1 + L(V)) \quad (c1)$$

$$\leq L(V) + \log_2(1 + L(V)) + \log_2 e. \quad (c2)$$

(d) (5 pts) With $V$ as above, show that if $U$ is a random variable taking values in $\{1, 2, \ldots\}$ for which $L(U) = L(V)$, then

$$H(U) \leq \sum_i \Pr(U = i) \log_2 \frac{1}{Q(i)} \quad (d1)$$

$$= \sum_i \Pr(V = i) \log_2 \frac{1}{Q(i)} \quad (d2)$$

$$= H(V).$$

Order the set of binary strings in increasing length ($\lambda$, 0, 1, 00, 01, \ldots), with $\lambda$ denoting the null string. Note that $[\log_2 i]$ is the length of the $i$'th string in this list.

(e) (5 pts) Suppose $U$ is a random variable taking values in $\{1, 2, \ldots\}$ with distribution $P$, and suppose $P(1) \geq P(2) \geq \ldots$. Find the non-singular code $C$ with smallest possible $E[\text{length}(C(U))]$, and express length $C(u)$ in terms of $\log_2 u$.

(f) (5 pts) For any random variable $U$ taking values in $\{1, 2, \ldots\}$, let

$$L^* = \min_{C: \text{non-singular}} E[\text{length}(C(U))].$$

Show that $H(U) \leq L^* + \log(1 + L^*) + \log_2 e$.  

3
Problem 3. (30 points) Consider the following variation on the Lempel-Ziv algorithm to encode an infinite sequence $u_1u_2 \ldots$ from an alphabet $U$.

1. Set the dictionary $\mathcal{D} = U$. Denote the dictionary entries as $d(0), \ldots, d(s-1)$, with $s = |U|$ being the size of the dictionary. Set $i = 0$ (the number of input letters read so far).

2. Find the largest $l$ such that $w = u_{i+1} \ldots u_{i+l}$ is in $\mathcal{D}$.

3. With $0 \leq j < s$ denoting the index of $w$ in $\mathcal{D}$, output the $\lceil \log_2 s \rceil$ bit binary representation of $j$.

4. Add the word $wu_{i+l+1}$ to $\mathcal{D}$, i.e., set $d(s) = wu_{i+l+1}$, and increment $s$ by $1$. Increment $i$ by $l$. Goto step 2.

For example, with $U = \{a, b\}$, the input string $abbbbaaab \ldots$ will lead to the execution steps:

<table>
<thead>
<tr>
<th>$\mathcal{D}$ at 2</th>
<th>$w$</th>
<th>output at 3</th>
<th>added-word at 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>a b</td>
<td>a</td>
<td>0</td>
<td>ab</td>
</tr>
<tr>
<td>a b ab</td>
<td>b</td>
<td>01</td>
<td>bb</td>
</tr>
<tr>
<td>a b ab bb</td>
<td>bb</td>
<td>11</td>
<td>bbb</td>
</tr>
<tr>
<td>a b ab bb bbb</td>
<td>b</td>
<td>001</td>
<td>ba</td>
</tr>
<tr>
<td>a b ab bb bbb ba</td>
<td>a</td>
<td>000</td>
<td>aa</td>
</tr>
<tr>
<td>a b ab bb bbb ba aa</td>
<td>aa</td>
<td>110</td>
<td>aab</td>
</tr>
</tbody>
</table>

(a) (5 pts) Can the decoder reconstruct the input sequence $u_1u_2 \ldots$ from the output of the algorithm? If so, how? (The crucial difficulty is that the description of $w$ in step 3 does not determine the word added to the dictionary in step 4.)

(b) (5 pts) The algorithm parses the sequence $u_1u_2 \ldots$ into a sequence of words $w_1w_2 \ldots$, (the $w$’s found in step 2). Show that a word $w$ can appear at most $|U|$ times in the parsing.

(c) (5 pts) Suppose $u^n = u_1 \ldots u_n$ is parsed into $m(u^n)$ words $w_1 \ldots w_m$ by the algorithm. Show that for any $k \geq 1$

$$n \geq k[m(u^n) - F(k)],$$

where $F(k) = |U| \sum_{i=1}^{k-1} |U|^i$.

(d) (5 pts) Show that $\lim_{n \to \infty} m(u^n)/n = 0$.

(e) (5 pts) Show that after reading $u^n$ the algorithm outputs fewer than $m(u^n)\lceil \log_2 |U| + m(u^n) \rceil$ bits.

Let $L(m, k)$ denote the minimum possible total length of a collection of $m$ binary strings where no string appears more than $k$ times.

(f) (5 pts) Show that if $u^n$ is fed to an information lossless finite state machine with $s$ states, then the machine outputs at least $L(m(u^n), s^2 |U|)$ bits.