PROBLEM 1. (a) We can write the following chain of inequalities:

\[
Q^n(x) = \prod_{i=1}^{n} Q(x_i) = 2 \prod_{a \in A} Q(a)^{N(a|x)} \geq \prod_{a \in A} 2^{n P_a(x) \log Q(a)}
\]

(b) Upper bound: We know that

\[
\sum_{k=0}^{n} \binom{n}{k} p^k (1-p)^{n-k} = 1.
\]

Consider one term and set \( p = k/n \). Then,

\[
1 \geq \binom{n}{k} \left( \frac{k}{n} \right)^k \left( 1 - \frac{k}{n} \right)^{n-k} = \binom{n}{k} 2^{n \left( \frac{k}{n} \log \frac{k}{n} + \frac{n-k}{n} \log \frac{n-k}{n} \right)} = \binom{n}{k} 2^{-nh_2 \left( \frac{k}{n} \right)}
\]

Lower bound: Define \( S_j = \binom{n}{j} p^j (1-p)^{n-j} \). We can compute

\[
\frac{S_{j+1}}{S_j} = \frac{n - j - 1}{j + 1} \frac{p}{1 - p}.
\]

One can see that this ratio is a decreasing function in \( j \). It equals 1, if \( j = np + p - 1 \), so \( \frac{S_{j+1}}{S_j} < 1 \) for \( j = B_{np} + pC \) and \( \frac{S_{j+1}}{S_j} \geq 1 \) for any smaller \( j \). Hence, \( S_j \) takes its maximum value at \( j = B_{np} + pC \), which equals \( k \) in our case. From this we have that

\[
1 = \sum_{j=0}^{n} \binom{n}{j} p^j (1-p)^{n-j} \leq (n + 1) \max_{j} \binom{n}{j} p^j (1-p)^{j} \leq (n + 1) \binom{n}{k} \left( \frac{k}{n} \right)^k \left( 1 - \frac{k}{n} \right)^{n-k} = \binom{n}{k} 2^{-nh_2 \left( \frac{k}{n} \right)}.
\]

The second equality comes from the derivation we had when proving the upper bound.

PROBLEM 2. Upon noticing \( 0.9^6 > 0.1 \), we obtain \( \{1, 01, 001, 0001, 00001, 000001, 0000001, 0000000\} \) as the dictionary entries.
Problem 3. Since the words of a valid and prefix condition dictionary reside in the leaves of a full tree, the Kraft inequality must be satisfied with equality: Consider climbing up the tree starting from the root, choosing one of the $D$ branches that climb up from a node with equal probability. The probability of reaching a leaf at depth $l_i$ is then $D^{-l_i}$. Since the climbing process will certainly end in a leaf, we have

$$1 = \Pr(\text{ending in a leaf}) = \sum_i D^{-l_i}.$$ 

If the dictionary is valid but not prefix-free, by removing all words that already have a prefix in the dictionary we would obtain a valid prefix-free dictionary. Since this reduced dictionary would satisfy the Kraft inequality with equality, the extra words would cause the inequality to be violated.

Problem 4.

(a) Let $I$ be the set of intermediate nodes (including the root), let $N$ be the set of nodes except the root and let $L$ be the set of all leaves. For each $n \in L$ define $A(n) = \{m \in N : m$ is an ancestor of $n\}$ and for each $m \in N$ define $D(m) = \{n \in L : n$ is a descendant of $m\}$. We assume each leaf is an ancestor and a descendant of itself. Then

$$E[\text{distance to a leaf}] = \sum_{n \in L} P(n) \sum_{m \in A(n)} d(m) = \sum_{m \in N} d(m) \sum_{n \in D(m)} P(n) = \sum_{m \in N} P(m)d(m).$$

(b) Let $d(n) = -\log Q(n)$. We see that $-\log P(n_j)$ is the distance associated with a leaf. From part (a),

$$H(\text{leaves}) = E[\text{distance to a leaf}] = \sum_{n \in N} P(n)d(n) = -\sum_{n \in N} P(n) \log Q(n) = -\sum_{n \in N} P(\text{parent of } n)Q(n) \log Q(n) = -\sum_{m \in I} P(m) \sum_{\text{n: n is a child of m}} Q(n) \log Q(n) = \sum_{m \in I} P(m)H_m'.$$

(c) Since all the intermediate nodes of a valid and prefix condition dictionary have the same number of children with the same set of $Q_n$, each $H_n = H$. Thus $H(\text{leaves}) = H \sum_{n \in I} P(n) = HE[L].$

Problem 5. (a)

$$E[F_n] = E[F_0X_0X_1 \ldots X_n] = F_0(E[X_1])^n = F_0(9/8)^n$$

We exploited the i.i.d. property of the sequence. One can see that $E[F_n] \to \infty$ with $n \to \infty$. 

2
(b) \[ l_n = E[\log_2 F_n] = E[\log_2 (F_0 X_0 X_1 \ldots X_n)] = E \left[ \log_2 F_0 + \sum_{i=1}^{n} \log_2 X_i \right] = \]
\[ = E[\log_2 F_0] + nE[\log_2 X_1] = \log_2 F_0 - \frac{n}{2}. \quad (4) \]

(c) It concentrates around $2^{l_n}$. $F_n$ in itself is not a sum of i.i.d. variables. Taking its logarithm results such a sum, so the law of large numbers applies.

\[ \log_2 F_n = \log_2 F_0 + \sum_{i=1}^{n} \log_2 X_i \rightarrow \log_2 F_0 + nE[\log X_1] = \log_2 F_0 - \frac{n}{2}. \]

(d) From the previous result it follows that although it seems appealing that the expected value of our fortune goes to infinity, it actually converges to 0 (very rapidly).

(e) We can equivalently say that instead of playing $n$ times, we create $2^n$ portions of our initial money, \(^n\text{C}_i\) portions of size $F_0 r^{n-i}(1-r)^i$, for all $i = 0, \ldots, n$. Then we bet $i$ times every $F_0 r^{n-i}(1-r)^i$ portion. We have seen that the more we play, the more we lose, so we should give smaller portions to large $i$ values. The best is to set $i = 0$ for all our money, that is $r = 1$, i.e. we don’t play at all.