Student Number: [redacted] Name: Bryan Hoang

1. (20 points)

(a) **Answer:**

Proof. We have

$$\begin{aligned} 2^{-l(j)} &\leq p(j) < 2^{-l(j)+1} \\ \Rightarrow -l(j) &\leq \log_2 p(j) < -l(j) + 1 \\ \Rightarrow l(j) &\geq -\log_2 p(j) > l(j) - 1. \end{aligned}$$

Thus, $l(j)$ is the lowest integer that is greater than or equal to $-\log_2 p(j)$. Therefore, we can conclude that $l(j) = \lceil -\log_2 p(j) \rceil$, so the expected code length satisfies $L(C) \leq H(X) + 1$. □

(b) **Answer:**

Proof. Let's prove the result by contradiction. First, suppose that C is not a prefix code. That is, let $i, j \in \mathcal{X} : i < j$ and assume that $C(i)$ is a prefix of $C(j)$, i.e., $C(i)$ appears in the first $l(i)$ bits of $C(j)$. Then $\hat{F}(i)$ and $\hat{F}(j)$ also share the first $l(i)$ bits. Thus, we have

$$\hat{F}(j) - \hat{F}(i) < 2^{-l(i)}. \tag{1}$$

But we also have

$$\begin{aligned} \hat{F}(j) - \hat{F}(i) &= \sum_{k=i}^{j-1} p(k) && \text{by the definition of } \hat{F}(j) \\ &\geq p(i) \\ &\geq 2^{-l(i)} && \text{by the definition of } l(i). \end{aligned} \tag{2}$$

With (1) and (2), we have a contradiction. Therefore, C is a prefix code. □