Solution for the exercise sheet n° 3 :

3-1.

(a) The code $\{\mathcal{C}(A) = 0, \mathcal{C}(B) = 0\}$ is singular. It is neither uniquely decodable nor instantaneous.

(b) The code $\{\mathcal{C}(A) = 0, \mathcal{C}(B) = 010, \mathcal{C}(C) = 01, \mathcal{C}(D) = 10\}$ is trivially nonsingular. It is not instantaneous, because the first codeword "0" is a prefix of the second one "01". The code is not uniquely decodable : For example, the sequence 010 can be associated to three distinct codewords B, AD or CA.

(c) The code $\{\mathcal{C}(A) = 10, \mathcal{C}(B) = 00, \mathcal{C}(C) = 11, \mathcal{C}(D) = 110\}$ is trivially nonsingular. It is not instantaneous : the third codeword "11" is a prefix of the last one "110". We will prove that the code is uniquely decodable. To decode we look at the first two bits of the sequence. If the first two bits are "00" then one decodes a B, if they are "10" one decodes an A. However, for "11" we do not know yet if it is a C or a D. To distinguish one needs to count the zeros after the first two "1" :

- If the number of zeros is even ($2k$), then the first two bits are a C (followed by k B's).
- If the number of zeros is odd ($2k+1$), then the first two bits are a D (followed by k B's).

One continues decoding the rest by repeating this method.

(d) The code $\{\mathcal{C}(A) = 0, \mathcal{C}(B) = 10, \mathcal{C}(C) = 110, \mathcal{C}(D) = 111\}$ is trivially nonsingular. It is instantaneous because no codeword is a prefix of another one. It is uniquely decodable.

3-2.

An instantaneous code satisfies the Kraft inequality : $\sum_i^m D^{-l_i} \leq 1$.
If $m = 6$ and lengths $\{l_i\} = \{1, 1, 1, 2, 2, 3\}$ the Kraft inequality reads : $3D^{-1} + 2D^{-2} + D^{-3} \leq 1$. Since D must be a minimal positive integer number satisfying inequality we have $D = 4$, that can be verified by testing the Kraft inequality for $D = 1, D = 2, D = 3, D = 4$, and $D = 5$.

This code is not optimal, because there is one with codeword lengths $\{l'_i\} = \{1, 1, 1, 2, 2, 2\}$. For instance $\{\mathcal{C}(x_1) = 0, \mathcal{C}(x_2) = 1, \mathcal{C}(x_3) = 2, \mathcal{C}(x_4) = 30, \mathcal{C}(x_5) = 31, \mathcal{C}(x_6) = 32\}$.

Remark : If $D = 4$ and amount of codewords is 6, an optimal code does not saturate the Kraft inequality. However, a binary code does. This can be verified by constructing a tree.

3-3.

(a) For the random variable X with letters $\{A, B, C, D\}$ which appear with probabilities $\{\frac{1}{10}, \frac{2}{10}, \frac{3}{10}, \frac{4}{10}\}$, we construct a binary Huffman code :

$$\left\{A, \frac{1}{10} \mid B, \frac{2}{10} \mid C \mid \frac{3}{10} \mid D, \frac{4}{10}\right\} \Rightarrow \begin{array}{rcl} A & = & 0 \\ B & = & 1 \end{array}$$

$$\left\{AB, \frac{3}{10} \mid C, \frac{3}{10} \mid D, \frac{4}{10}\right\} \Rightarrow \begin{array}{rcl} AB & = & 0 \\ C & = & 1 \end{array}$$

$$\left\{ABC, \frac{6}{10} \mid D, \frac{4}{10}\right\} \Rightarrow \begin{array}{rcl} ABC & = & 0 \\ D & = & 1 \end{array}$$

This leads to the code : $\{\mathcal{C}(A) = 000, \mathcal{C}(B) = 001, \mathcal{C}(C) = 01, \mathcal{C}(D) = 1\}$.

(b) We remark that if the cardinality of the alphabet $d \geq 3$, it is not always possible to have enough symbols to combine in packets of d. In this case, it is necessary to add additional symbols which appear with zero probabilities. Since at each iteration the number of symbols is reduced by $d-1$, we should have in total $1 + k(d-1)$ symbols, where k is the depth in the tree. Let us construct now a ternary Huffman code for X. For $d = 3$, we should have an odd number of symbols. If we add the symbol E with zero probability, we get :

$$\begin{aligned} \left\{A, \frac{1}{10} \mid B, \frac{2}{10} \mid C, \frac{3}{10} \mid D, \frac{4}{10} \mid E, \frac{0}{10}\right\} \Rightarrow A &= 0 \\ B &= 1 \\ E &= 2 \end{aligned}$$

$$\begin{aligned} \left\{ABE, \frac{3}{10} \mid C, \frac{3}{10} \mid D, \frac{4}{10}\right\} \Rightarrow ABE &= 0 \\ C &= 1 \\ D &= 2 \end{aligned}$$

This leads to the code : $\{\mathcal{C}(A) = 00, \mathcal{C}(B) = 01, \mathcal{C}(C) = 1, \mathcal{C}(D) = 2\}$.

(c) A Shannon code, associates a length $l_i = \left\lceil \log_2 \frac{1}{p_i} \right\rceil$ with each symbol x_i, that is emitted by the source $\{x_i, p_i\}$. Thus, $\{l(A) = 4, l(B) = 3, l(C) = 2, l(D) = 2\}$. To construct such a code one can take the previous binary Huffman code. For example, a possible Shannon code is $\{\mathcal{C}(A) = 0000, \mathcal{C}(B) = 001, \mathcal{C}(C) = 01, \mathcal{C}(D) = 11\}$.

The expected length of the Shannon code is $\bar{l}_S = 2.4$ bits; the expected length of the Huffman code is $\bar{l}_H = 1.9$ bits.

The bound on the optimal code length reads $H(X) = -\sum_{i=1}^{4} p_i \log_2 p_i = 1.85$ bits. Thus, we verified the inequality $H(X) \leq \bar{l}_H \leq \bar{l}_S < H(X) + 1$.

3-4.

(a) Let X take the values $\{A, B, C, D\}$ with the given probabilities $\{\frac{1}{3}, \frac{1}{3}, \frac{1}{4}, \frac{1}{12}\}$. We construct a binary Huffman code for X :

$$\begin{aligned} \left\{A, \frac{1}{3} \mid B, \frac{1}{3} \mid C, \frac{1}{4} \mid D \mid \frac{1}{12}\right\} \Rightarrow C &= 1 \\ D &= 0 \end{aligned}$$

$$\begin{aligned} \left\{A, \frac{1}{3} \mid B, \frac{1}{3} \mid CD, \frac{1}{3}\right\} \Rightarrow B &= 1 \\ CD &= 0 \end{aligned}$$

$$\begin{aligned} \left\{BCD, \frac{2}{3} \mid A, \frac{1}{3}\right\} \Rightarrow BCD &= 1 \\ A &= 0 \end{aligned}$$

This gives the code $\{\mathcal{C}(A) = 0, \mathcal{C}(B) = 11, \mathcal{C}(C) = 101, \mathcal{C}(D) = 100\}$. Alternatively, by regrouping A and B in the second step, one obtains the Huffman code $\{\mathcal{C}(A) = 11, \mathcal{C}(B) = 10, \mathcal{C}(C) = 01, \mathcal{C}(D) = 00\}$ with the same expected length.

(b) For the Shannon code we obtain $\{l(A) = 2, l(B) = 2, l(C) = 2, l(D) = 4\}$. To construct this code we can take the previous Huffman code. For example, one of possible Shannon codes reads $\{\mathcal{C}(A) = 11, \mathcal{C}(B) = 10, \mathcal{C}(C) = 01, \mathcal{C}(D) = 0000\}$.

The expected length of the Shannon code is $\bar{l}_S = 2.17$ bits; the expected length of the Huffman code is $\bar{l}_H = 2$ bits.

The bound on the optimal code length reads $H(X) = -\sum_{i=1}^{4} p_i \log_2 p_i = 1.86$ bits. Thus, we have verified the inequality : $H(X) \leq \bar{l}_H \leq \bar{l}_S < H(X) + 1$. However, the property $\bar{l}_H \leq \bar{l}_S$ does not imply that the length of all codewords of the Shannon code are bigger than the ones of the Huffman code (see previous examples).

3-5.

(a) Alice flips the coin k times and obtains "head" at the k-th try. The expected length of the naive code reads

$$\bar{l}_n = \sum_{k=1}^{\infty} kp(1-p)^{k-1} = \frac{1}{p}.$$

The entropy of the random variable k is given by

$$\begin{aligned}
H(k) &= -\sum_{k=1}^{\infty} (1-p)^{k-1} p \ \log_2 \big((1-p)^{k-1} p\big) \\
&= -p \log_2(1-p) \sum_{k=1}^{\infty} (k-1)(1-p)^{k-1} - p \log_2 p \sum_{k=1}^{\infty} (1-p)^{k-1} \\
&= -\big[p \log_2(1-p)\big] \frac{1-p}{p^2} - \big[p \log_2 p\big] \frac{1}{p} \\
&= -\frac{1}{p} \big(p \ \log_2 p + (1-p) \ \log_2(1-p)\big) \\
&= \frac{1}{p} H(p, 1-p)
\end{aligned}$$

Since $H(p, 1-p) \leq 1$, we have $H(k) \leq \frac{1}{p} = \bar{l}_n$.
The naive code is optimal [i.e. $\bar{l}_n = H(k)$] if $p = 1/2$.

(b) In the limit $p \to 0$ the expected length of the naive code diverges to infinity. The expected length of the Shannon code is approximately the entropy : $\bar{l}_S \simeq H(k)$ (remember the inequality $H(k) \leq \bar{l}_S < H(k) + 1$). In the limit $p \to 0$ we obtain

$$\lim_{p \to 0} \bar{l}_S \simeq \lim_{p \to 0} H(k) \simeq \lim_{p \to 0} \frac{H(p, 1-p)}{p} \simeq \lim_{p \to 0} \log_2 \frac{1}{p},$$

which also diverges to infinity, but as a logarithmic of that for naive code : $\bar{l}_n = \frac{1}{p} \simeq 2^{\bar{l}_S}$. Thus, there is an exponential gain with respect to the naive code.