CPSC 530/630
<u>Assignment 1 Solution</u>

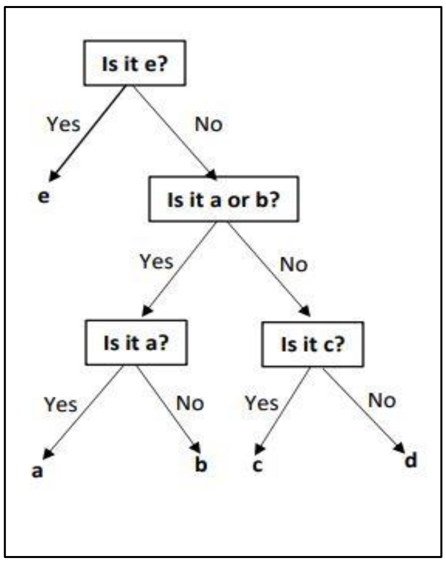
Q2:

1) $H(V) = -[2*(0.1*\log_2(0.1)) + 2*(0.05*\log_2(0.05)) + (0.7*\log_2(0.7))] = \mathbf{1.46}$ bits

2) $Pr(X=0) = \mathbf{0.3}$
$Pr(X=1) = \mathbf{0.7}$
$H(X) = (0.3*-\log_2(0.3)) + (0.7*-\log_2(0.7)) = \mathbf{0.88}$ bits

$Pr(Y=0) = \mathbf{0.15}$
$Pr(Y=1) = \mathbf{0.15}$
$Pr(Y=2) = \mathbf{0.7}$
$H(Y) = -[2*(0.15*\log_2(0.15)) + (0.7*\log_2(0.7))] = \mathbf{1.18}$ bits

Comparison:
V had the highest entropy, followed by Y, then X. This is because all three had the same uncertainty for e, Pr(V = e) = 0.7, but for the remaining 0.3; X had no uncertainty, so had the lowest entropy, Y had some uncertainty with the two options of 0.15, and V had the most uncertainty with the four options of 0.1, 0.1, 0.05, and 0.05, giving it the highest entropy.

3)


<u>Expected Number of Questions for V</u>:
= (0.1x3) + (0.1x3) + (0.05x3) + (0.05x3) + (0.7x1)
= **1.6** questions

<u>Relationship to H(V):</u>

$H(V) = 1.46 \leq 1.6 \leq H(V)+1 = 2.46$

4) $$E[X] = \sum_j x_j Pr(x_j) = (0 * 0.3) + (1 * 0.7) = \mathbf{0.7}$$
$$E[Y] = \sum_j y_j Pr(y_j) = (0 * 0.15) + (1 * 0.15) + (2 * 0.7) = \mathbf{1.55}$$

5) $\mathbf{Z_1 = Y + 1 \pmod 3 = \{1, 2, 0\}}$
$Pr(Z_1=0) = Pr(Y=2) = 0.7$
$Pr(Z_1=1) = Pr(Y=0) = 0.15$
$Pr(Z_1=2) = Pr(Y=1) = 0.15$
$\boldsymbol{H(Z_1)} = \mathbf{1.18}$ bits

$\mathbf{Z_2 = Y^2 \pmod 3 = \{0, 1\}}$
$Pr(Z_2=0) = Pr(Y=0) = 0.15$
$Pr(Z_2=1) = Pr(Y=1) + Pr(Y=2) = 0.85$

$\boldsymbol{H(Z_2)} = \mathbf{0.6098}$ bits

6) For Z_1, the probability distributions didn’t change by doing re-ordering. So, the total amount of uncertainty within the system remained the same as Y.

For Z_2, while mapping Y to Z_2 we are neglecting Y=1 and Y=2, and pretending they are the same thing. It reduces the amount of uncertainty, thus the amount of information is less here compared to Y.

Q3

1. 14 a. $A_a = \{a, b, \cdots z, 0, 1 \cdots 9\}$ $|A_a| = 36$ 3

7 $H(S_a) = \log_2 |36^4 + 36^5 + 36^6| \approx 31.06$ bits 4

Assuming all choices are equally likely

b. F the set of 10 fruits

7 $\Rightarrow A_b = F \times F$ $|A_b| = 100$ 3

$\Rightarrow H(S_b) = \log_2 |100^3| \approx 19.93$ bits 4

2. 16 CPSC 530

a. For a password of length 4, one character is typed incorrectly

8 eg. 1 2 3 4

_ 2 3 4 *

1 _ 3 4

1 2 _ 4

1 2 3 _

The blank could be 35 possibilities (except ~~a sin~~ 1 in *)

So we could have $35 \times 4 + 1$ accepted passwords.

5 - character password can have $35 \times 5 + 1$ accepted entries. 2

6 - character password can have $35 \times 6 + 1$ accepted entries 2

The success chance of the Eve becomes

$$P_r = \frac{35 \times 4 + 1 + 35 \times 5 + 1 + 35 \times 6 + 1}{36^4 + 36^5 + 36^6} = \frac{528}{36^4 + 36^5 + 36^6}$$

$$H(S_a) = -\log_2 \frac{528}{36^4 + 36^5 + 36^6} \approx 22.02 \text{ bits}$$ 1

CPSC 630

9 For a password of length 4

i) a character is typed incorrectly

the number of accepted passwords is 35×4

ii) a character shorter

the # of accepted passwords is 4

iii) one character longer

e.g. ↑ 1 ↑ 2 ↑ 3 ↑ 4 ↑

Insert one character at any of the 5 positions

the # of accepted passwords is 36×5

In total the # of accepted passwords is

$$1 + 35 \times 4 + 4 + 36 \times 5$$ 2'

For a password of length 5, the number of accepted passwords is

$$1 + 35 \times 5 + 5 + 36 \times 6$$ 2'

For a password of length 6, the number of accepted passwords is

$$1 + 35 \times 6 + 6 + 36 \times 7$$ 2'

The success chance of the Eve becomes

$$Pr = \frac{1+35\times4+4+36\times5+1+35\times5+5+36\times6+1+35\times6+6+36\times7}{36^4+36^5+36^6}$$

$$= \frac{1191}{36^4+36^5+36^6}$$

$H(S_a) = -\log_2 Pr \approx 20.84$ bits 1'

2b. $Pr = \dfrac{6}{10^6}$ 3'

Hint: a tuple of 3 elements can be written in 6 different ways.

$H(S_b) = -\log_2 \dfrac{6}{10^6} \approx 17.35$ bits 3'

In 2(a), By changing the verification algorithm, the success chance of the adversary guessing the correct password increases, while the entropy of the system decreases. 1'

In 2(b). the security level of the system decreases. 2

More on Q3.2 (b)

Actually, not all passwords have 6 permutations. For example, when all three pairs are identical. It can be only written in 1 way. However doing exact counting could be complicated.

If we assumed 6 permutations, and divided the number by 6. It will be a lower bound security - that is you consider the worst case.

If we want to have a more precise estimation of uncertainty, then the exact counting process is shown in below *:

For the modified verification algorithm, we simply count the number of unordered sets $\{(x_1, y_1), (x_2, y_2), (x_3, y_3)\}$.

There are 100 such sets where all three pairs are identical, $\binom{100}{2} \times 2 = 9900$ sets where exactly 2 two of the three pairs are identical, and $\binom{100}{3} = 161700$ pairs where all 3 three pairs are distinct. Thus, there are effectively 171700 passwords in total. There entropy of this modified system is then log(171700) = 17.4 bits.

*This solution is from Janet Leahy

Q4:

1) **H($\mathcal{A}$)** = **-** [(0.5 x $\log_2(0.5)$) + (0.33 x $\log_2(0.33)$) + (0.083 x $\log_2(0.083)$) + (0.083 x $\log_2(0.083)$)]
= **1.63** bits/letter

2) Optimal Code, C = {0, 10, 110, 111} or {1, 01, 001, 000}
L_{EXP} = (1 x 0.5) + (2 x 0.33) + (3 x 0.083) + (3 x 0.083) = **1.67** binary digits

3) Alphabet = {AA,BB,CC,DD,AB,BA,CD,DC,AC,CA,BD,DB,AD,DA,BC,CB}

Code:	Length:	Probability
C(AA) = 01	l(AA) = 2	Pr(AA) = 1/4
C(AB) = 000	l(AB) = 3	Pr(AB) = 1/6
C(BA) = 001	l(BA) = 3	Pr(BA) = 1/6
C(BB) = 101	l(BB) = 3	Pr(BB) = 1/9
C(AC) = 1101	l(AC) = 4	Pr(AC) = 1/24
C(CA) = 1110	l(CA) = 4	Pr(CA) = 1/24
C(AD) = 1111	l(AD) = 4	Pr(AD) = 1/24
C(DA) = 10000	l(DA) = 5	Pr(DA) = 1/24
C(BD) = 10001	l(BD) = 5	Pr(BD) = 1/36
C(DB) = 11000	l(DB) = 5	Pr(DB) = 1/36
C(BC) = 11001	l(BC) = 5	Pr(BC) = 1/36
C(CB) = 10010	l(CB) = 5	Pr(CB) = 1/36
C(CC) = 1001110	l(CC) = 7	Pr(CC) = 1/144
C(DD) = 1001111	l(DD) = 7	Pr(DD) = 1/144
C(CD) = 1001100	l(CD) = 7	Pr(CD) = 1/144
C(DC) = 1001101	l(DC) = 7	Pr(DC) = 1/144

Expected length of code:
L_{EXP} = (2/4) + (3/6) + (5/24) + (4/24) + (3/6) + (3/9) + (5/36) + (5/36) + (4/24) + (5/36) + (7/144) + (7/144) + (4/24) + (5/36) + (7/144) + (7/144) = (79/24) = **3.29** bits /2 symbols
= **1.65** bits/symbol

4) Q2 Code length, $L_2 = 23$
Expected number of binary digits = 23/12 = 1.9167

Q3 Code length. $L_3 = 24$
Expected number of binary digits = 24/12 = 2

Even though $Lexp\_{q3} < Lexp\_{q2}$, we can see that using encoding in Q4.3 have larger expected number of binary digits than Q4.2, that is used for sending each source output.