The University of Newcastle
School of Electrical Engineering and Computer Science

COMP3260/COMP6360 Data Security
Workshop Week 3 Solutions
8th and 10th March 2021

1. Using extended Euclid's algorithm, find the solution to the equation $17x \bmod 100 = 1$ in the range [0, 99].

Solution:
We need to find the multiplicative inverse of 17 mod 100 using extended Euclid's algorithm. We already know that the result is 53 (see exercise 9.c from last week).

```
Input: a, n
Output: None
inv(a,n) {
      g[0] = n; g[1] = a
      u[0] = 1; u[1] = 0
      v[0] = 0; v[1] = 1
      i = 1
      while (g[i] ≠  0) // “g[i] = u[i]n + v[i]a”
      {
            y = g[i-1] / g[i]  //integer division
            g[i+1] = g[i-1] – y × g[i]
            u[i+1] = u[i-1] – y × u[i]
            v[i+1] = v[i-1] – y × v[i]
            i = i +1
      }
      if v[i-1] ≥ 0 then  return v[i-1] else return v[i-1] + n
}
```

i	y	u	v	g
0		1	0	100
1		0	1	17
2	5	1	-5	15
3	1	-1	6	2
4	7	8	-47	1
5	2	-17	100	0

$x = -47 \bmod 100 = 53$ (Remember to add n when $x_0<0$)

Note that u_4 gives is the multiplicative inverse of 100 mod 17, that is, the solution of $100x \bmod 17 = 1$.

Checking: $100 \times 8 \bmod 17 = 15 \times 8 \bmod 17 = 120 \bmod 17 = 1$

2. Using Euler's theorem and fast exponentiation, solve the following equation for x in the range [0, n-1].

a) 5x mod 17 = 1
b) 19x mod 26 = 1
c) 17x mod 100 = 1
d) 2x mod 57 =1

Solution:

a) We can use Euler's theorem:

$5^{\Phi(17)-1} \bmod 17 = x$
$5^{16-1} \bmod 17 = x$

Using fast exponentiation we get

$x = 5^{15} \bmod 17 = 5 \times 5^{14} \bmod 17$
$= 5 \times 25^{7} \bmod 17 = 5 \times 8^{7} \bmod 17$
$= 5 \times 8 \times 8^{6} \bmod 17 = 6 \times 8^{6} \bmod 17$
$= 6 \times 64^{3} \bmod 17 = 6 \times 13^{3} \bmod 17$
$= 6 \times 13 \times 13^{2} \bmod 17 = 10 \times 13^{2} \bmod 17$
$= 10 \times 16 \bmod 17 =$ **7**

b) Euler's theorem:

$19\phi^{(26)-1} \bmod 26 = x$
$26 = 2 \times 13$
$\phi(26) = (2-1) \times (13-1) = 12$

$x = 19^{12-1} \bmod 26 = 19^{11} \bmod 26$
$= 19 \times 19^{10} \bmod 26$
$= 19 \times 361^{5} \bmod 26 = 19 \times 23^{5} \bmod 26$
$= 19 \times 23 \times 23^{4} \bmod 26 = 21 \times 23^{4} \bmod 26$
$= 21 \times 9^{2} \bmod 26$
$= 21 \times 81 \bmod 26 = 21 \times 3 \bmod 26 = 63 \bmod 26 =$ **11**

c) $100 = 2^{2} \times 5^{2}$

$\phi(100) = (2-1) \times 2 \times (5-1) \times 5 = 40$
$x = 17^{39} \bmod 100 = 17 \times 17^{38} \bmod 100 = 17 \times (17^{2})^{19} \bmod 100 =$
$= 17 \times 289^{19} \bmod 100 =$
$= 17 \times 89^{19} \bmod 100 = 17 \times 89 \times 89^{18} \bmod 100 = 13 \times (89^{2})^{9} \bmod 100 =$
$= 13 \times 7921^{9} \bmod 100 = 13 \times 21^{9} \bmod 100 = 13 \times 21 \times 21^{8} \bmod 100 =$
$= 73 \times (21^{2})^{4} \bmod 100 = 73 \times 41^{4} \bmod 100 = 73 \times (41^{2})^{2} \bmod 100 =$
$= 73 \times 81^{2} \bmod 100 = 73 \times 61 \bmod 100 =$ **53**

d) 2x mod 57 =1

$n = 57 = 3 \times 19$
$\phi(n) = 2 \times 18 = 36$

Using Euler's theorem we get
$x = 2^{35} \bmod 57 = 29$

Working:
2^{35} mod 57 = 2×2^{34} mod 57 = 2× $(2^2)^{17}$ mod 57 = 2×4^{17} mod 57 =
= 2×4×4^{16} mod 57 = 8× $(4^2)^8$ mod 57 = 8×16^8 mod 57 = 8× $(16^2)^4$ mod 57 =
=8×256^4 mod 57 = 8×28^4 mod 57 = 8×$(28^2)^2$ mod 57 = 8×784^2 mod 57 =
=8×43^2 mod 57 = 8×1849 mod 57 = 8×25 mod 57 = **29**

3. Find the inverse of 5 mod 31.

Solution:

n = 31; ϕ (n) = 30

Using Euler's theorem we get

x = $5^{\phi(31)-1}$ mod 31 = 5^{29} mod 31 = 25

Working:
5^{29} mod 31 = 5×5^{28} mod 31 = 5×$(5^2)^{14}$ mod 31 = 5×25^{14} mod 31 =
= 5× $(25^2)^7$ mod 31 = 5×625^7 mod 31 = 5×5^7 mod 31 = 5×5×5^6 mod 31 =
= 25×25^3 mod 31 = 25×25×25^2 mod 31 = 5×5 mod 31 = **25**

4. Find all solutions to the equation 15x mod 25 = 10 in the range [0, 24].

Solution:

gcd(15,25)=5

Since 5 divides 10, the equation 15x mod 25 = 10 has 5 solutions of the form
$x_t = (2x_0 + 5t)$ mod 25, t=0,1,2,3,4 where x_0 is the solution to 3x mod 5 =1.

We have x_0 = 2 and x_t = (4+5t) mod 25, t=0,1,2,3,4:
$x_1 = 4$
$x_2 = 9$
$x_3 = 14$
$x_4 = 19$
$x_5 = 24$

5. Consider GF(2^3) with the irreducible polynomial p(x)=1011 (x^3+x+1). Find additive and multiplicative inverses of all elements of this field.

Solution:

GF(2^3) consists of all polynomials with degree at most 2, that is, the following polynomials:
0 0 0
0 0 1
0 1 0
0 1 1
1 0 0
1 0 1
1 1 0
1 1 1

In GF(2^3), the additive inverse of a polynomial a is a polynomial b such that $a + b = 0\ 0\ 0$. We denote the additive inverse of a by $-a$. Multiplicative inverse of a polynomial a is a polynomial b such that $a \times b = 0\ 0\ 1$. We denote the multiplicative inverse of a by a^{-1}.

All elements of GF(2^3) have an additive inverse. For each a we obtain $-a$ by subtracting a from 0 0 0. Note that in GF(2^3), subtraction is equivalent to bitwise XOR. For example, for $a = 0\ 0\ 0$ we get:

```
 0 0 0
- 0 0 0
-------------
 0 0 0
```

and thus the additive inverse for $0\ 0\ 0$ is $0\ 0\ 0$.

For $a = 0\ 0\ 1$ we get

```
 0 0 0
- 0 0 1
-------------
 0 0 1
```

and thus the additive inverse for $0\ 0\ 1$ is $0\ 0\ 1$. In general, for each element of GF(2^3) we have $a = -a$.

All elements of GF(2^3) except $0\ 0\ 0$ have a multiplicative inverse. To find multiplicative inverse of a: $a^{-1} = a^{\Phi(p(x))-1}$ mod $p(x)$, where $\Phi(p(x)) = 7$ (see lecture notes).

For example, we obtain the multiplicative inverse of 0 0 1 as follows:
$a = 0\ 0\ 1$
$a^{-1} = 0\ 0\ 1^{7-1}$ mod $1011 = 0\ 0\ 1^{6}$ mod 1011
a^2:

```
   0 0 1
 × 0 0 1
 -----------
   0 0 1
  0 0 0
0 0 0
 ----------
0 0 0 0 1
```

Thus $a^2 = 0\ 0\ 1$. As $a = a^2$, we have $a = a^2 = a^4 = a^6 = a^{-1} = 0\ 0\ 1$
In the same way we find multiplicative inverses for all elements of GF(2^3). The results are summarized in the following table:

a	-a	a^{-1}
000	000	-
001	001	001
010	010	101

011	011	110
100	100	111
101	101	010
110	110	011
111	111	100

An alternative way to find multiplicative inverses of all elements of GF(2^3) is to construct a multiplication table and read off the multiplicative inverses as a row and a column having 0 0 1 in their intersection:

	000	001	010	011	100	101	110	111
000	000	000	000	000	000	000	000	000
001	000	***001***	010	011	100	101	110	111
010	000	010	100	110	011	***001***	111	101
011	000	011	110	101	111	100	***001***	010
100	000	100	011	111	110	010	101	***001***
101	000	101	***001***	100	010	111	011	110
110	000	110	111	***001***	101	011	010	100
111	000	111	101	010	***001***	110	100	011

6. Evaluate complexity of algorithm for fast exponentiation.

Solution:

To evaluate the complexity of an algorithm, we first need to identify a barometer statement that is executed at least as many times as any other statement in the algorithm. We can then consider the worst case and provide O (big Oh, of order at most) for the number of times the barometer statement is executed.

```
Input: a, z, n
Output: None
fastexp(a,z,n) {
      // x = $a^z$ mod n
      x = 1
      while (z ≠ 0) {
            while (z mod 2 == 0) {
                  z= z/2
                  a = a*a mod n
            }
            z = z-1
            x = x*a mod n
      }
      return x
}
```

We choose the comparison **z mod 2 == 0** in the second while loop as the barometer. statement. This statement is executed once for every 0 and twice for every 1 in the binary representation of the exponent z. In the worst case when we have all 1′s, the number of time the barometer statement is executed is $2\times \lceil \log_2 z \rceil = O(\log z)$. Thus we say that the number of steps taken by the algorithms in the worst case is of order at most log z. (Note that we can use the same argument to show the tight bound for both worst and best case: the number of time the barometer statement is executed in the worst case is $2\times\lceil \log_2 z\rceil$, and in the best case is $\lceil \log_2 z\rceil$; since $2\times\lceil \log_2 z\rceil = \Theta(\log z)$, and also $\lceil \log_2 z\rceil = \Theta(\log z)$, we have that the tight bound for worst, best and average case is $\Theta(\log z)$).

7. Evaluate complexity of Euclid's algorithm for finding the greatest common divisor of two integers.

Solution:

We choose the comparison $g_i \neq 0$ as a barometer statement – see the algorithm bellow.

```
Input: a, n
Output: None
gcd(a,n) {
      g[0] = n
      g[1] = a
      i = 1
      while (g[i] ≠  0){
            g[i+1] = g[i-1] mod  g[i]
            i = i +1
      }
      return g[i-1]
}
```

The barometer statement is executed k-1 times, where n is the index of g such that $g_k=0$ and g_0, g_1, ..., $g_{k-1} > 0$.

To evaluate k, we first observe that $g_i < g_{i-1}$ and $g_i \le g_{i-2} - g_{i-1}$, for any i>1. For g_i, g_{i+1}, and g_{i+2} where $i \ge 0$, we consider 2 cases:

1. $g_{i+1} \le g_i /2$; since $g_{i+2} < g_{i+1}$, we have $g_{i+2} < g_i /2$
2. $g_{i+1} > g_i /2$; since $g_{i+2} \le g_i - g_{i+1}$, we have $g_{i+2} < g_i /2$

Therefore we always have $g_{i+2} < g_i /2$ and thus the number of binary digits in g_{i+2} is at least one less than the number of binary digits in g_i. It follows that in the worst case $k - 1 \le 2\times\lceil \log_2 g_0\rceil = O(\log g_0)$.

8. Use the Theorem presented in the lecture (see bellow) to explore if there is a simple way to solve '*n* mod *d*' for d=2, 3, 4, 5, 6, 7, 8 and 9. For example, n mod 3 can be found by adding up all the decimal digits of n, and taking mod 3 of the sum.

Theorem: Let *a* and *b* be integers, and let *op* be one of the binary operators +, -, or *. Then *(a op b) mod n = [(a mod n) op (b mod n)] mod n*

Solution idea:

d=2; we have $10^0 \bmod 2 = 1$ and $10^k \bmod 2 = 0$ for $k > 0$. Then for any number x with decimal digits $x_k x_{k-1} x_{k-2} \ldots x_0$ we have

$$x \bmod 2 = (10^k x_k + 10^{k-1} x_{k-1} + \ldots + 10 x_1 + x_0) \bmod 2 =$$

$$= (10^k x_k \bmod 2 + 10^{k-1} x_{k-1} \bmod 2 + \ldots + 10 x_1 \bmod 2 + x_0 \bmod 2) \bmod 2$$

$= (0 \times x_k \bmod 2 + 0 \times x_{k-1} \bmod 2 + ... + 0 \times x_1 \bmod 2 + x_0 \bmod 2) \bmod 2$

$= (0 \bmod 2 + 0 \bmod 2 + ... + 0 \bmod 2 + x_0 \bmod 2) \bmod 2$

$= x_0 \bmod 2$

9. Let X be an integer variable represented with 32 bits. Suppose that the probability is ½ that X is in the range $[0, 2^8-1]$, with all such values being equally likely, and ½ that X is in the range $[2^8,2^{32}-1]$, with all such values being equally likely. Compute H(X).

Solution: There are 2^8 numbers in the range $[0, 2^8 – 1]$ and they are all equally likely; thus the probability for each such number is $1/2 \times 1/2^8 = 1/2^9$. Similarly, there are $2^{32} – 2^8$ numbers in the range $[2^8, 2^{32} – 1]$ and they are also all equally likely; thus the probability for each such number is $1/2 \times 1/(2^{32} – 2^8) = 1/(2^{33} – 2^9)$. Then entropy H(X) is:

$H(X) = \Sigma\, p(X) \log_2 1/p(X) = 2^8 \times 1/2^9 \times \log_2 2^9 + (2^{32} – 2^8) \times 1/(2^{33} – 2^9) \times \log_2 (2^{33} – 2^9) = 9/2 + 1/2 \times \log_2 (2^{33} – 2^9) \approx 9/2 + 1/2 \times \log_2 2^{33} = 9/2 + 33/2 = 42/2 = 21$ bits.

10. Let X be one of the 6 messages: A, B, C, D, E and F, where:
$p(A)=p(B)=p(C)=1/4$
$p(D)=1/8$
$p(E)=p(F)=1/16$
Compute H(X) and find an optimal binary encoding of the message.

Solution:

$H(X) = \Sigma\, p(X) \log_2 1/p(X) = 19/8$ bits

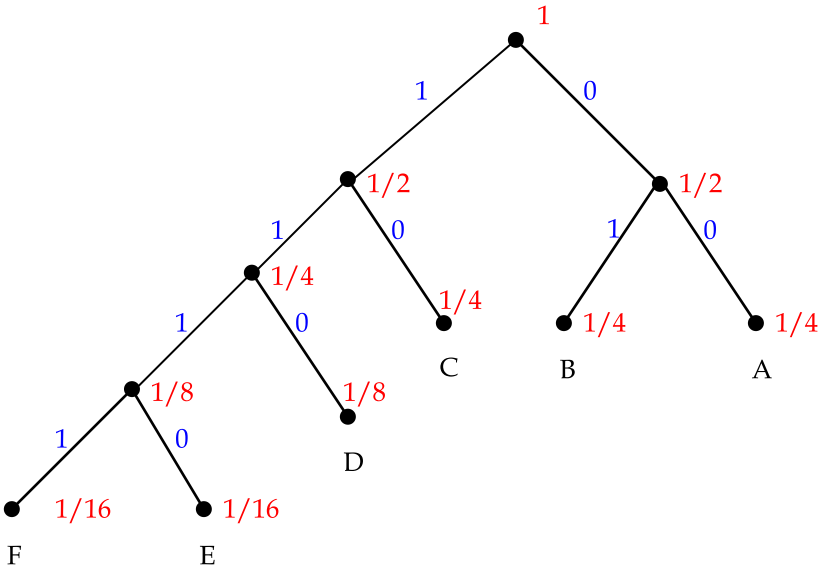
X	p(X)	1/p(X)	$\log_2 (1/p(X))$	$p(X) \log_2 (1/p(X))$
A	1/4	4	2	1/2
B	1/4	4	2	1/2
C	1/4	4	2	1/2
D	1/8	8	3	3/8
E	1/16	16	4	1/4
F	1/16	16	4	1/4

$H(X) = \Sigma\, p(X) \log_2 1/p(X) = 3 \times 1/4 \times \log_2 4 + 1/8 \times \log_2 8 + 2 \times 1/16 \times \log_2 16 = 6/4 + 3/8 + 8/16 = 19/8 = 2.375$ bits.

We now need to find an optimal encoding for these messages. We use Huffman code. We start to build a Huffman tree; we first insert a leaf for each message and we label it with the probability corresponding to that leaf. We then combine two nodes with the smallest probabilities by adding a parent node and connecting it to both nodes; the probability of the parent node is the sum of probabilities of the two nodes. We continue this process until we introduce a node with probability 1 – that is the root of the tree and we are done.

We then label edges of the Huffman tree with labels 0 and 1, such that from each internal node (that is, non-leaf), one edge is labelled 1 and the other edge is labelled 0 (note that this is a binary tree and so each internal tree has two edges connecting it to its children nodes). Then for each message, we find the encoding by reading off the labels of all the edges between the root and the leaf corresponding to that message.

A	1/4	1/4	1/4	1/4	1/2	1
B	1/4	1/4	1/4	1/4		
C	1/4	1/4	1/4	1/2	1/2	
D	1/8	1/8	1/4			
E	1/16	1/8				
F	1/16					



A=00, B=01, C=10, D=110, E=1110, F=1111.

11. Suppose there are 5 possible messages, A, B, C, D and E, with the probabilities p(A)= 0.5, p(B)= 0.3,p(C)= 0.1, p(D)= 0.05 and p(E)= 0.05. What is the expected number of bits needed to encode these messages in optimal encoding? (That is, find H(M).) Provide optimal encoding.

<u>**Solution:**</u>

$H(M) = \Sigma\, p(M) \log_2 1/p(M)$

$= 1/2 \log_2 2 + 3/10 \log_2 10/3 + 1/10 \log_2 10 + 2 * 1/20 \log_2 20$

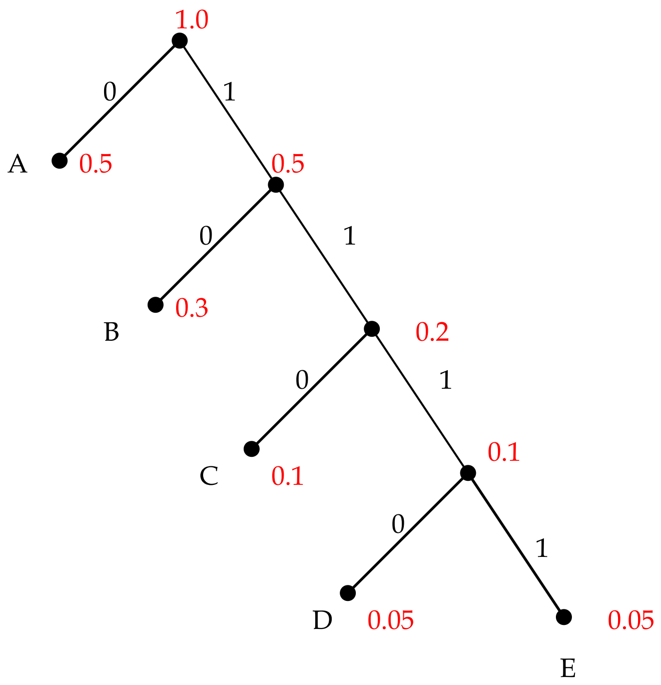
$= 1/2 + 3/10 (\log_2 10 - \log_2 3) + 1/10 \log_2 10 + 1/10(\log_2 10 - \log_2 2)$

$= 5/10 + 3/10 \log_2 10 – 3/10 \log_2 3 + 1/10 \log_2 10 + 1/10 \log_2 10 + 1/10$

$= 6/10 + 5/10 \log_2 10 – 3/10 \log_2 3$

$= 0.6 + 0.5 \log_2 10 – 0.3 \log_2 3$

$= 1.785$ Bits



Hence the encoding is
A = 0, B = 10, C = 110, D = 1110 and E = 1111
Is it optimal?

For each possible message we multiply the probability of the message occurring and the number of bits used to encode that message, and sum for all messages.

So the average number of bits N_{AVG} for the above encoding would be.

$N_{AVG} = (p(A)\times 1) + (p(B)\times 2) + (p(C)\times 3) + (p(D)\times 4) + (p(E)\times 4)$

$= (0.5\times 1) + (0.3\times 2) + (0.1\times 3) + 2(0.05\times 4) = 1.8$ Bits.

This is slightly higher than the entropy of 1.785 Bits.

12. Let M be a secret message revealing the recipient of a scholarship. Suppose there were one female applicant, Anne, and three male applicants, Bob, Doug and John. It was initially thought each applicant had the same chance of receiving scholarship; thus p(Anne) = p(Bob) = p(Doug) = p(John) = ¼. It was latter learned that the chances of a scholarship going to a female were ½. Letting S denote the message revealing the sex of the recipient, compute $H_S(M)$.

Solution:

Before the extra knowledge about the sex of the recipient, the probabilities and the entropy were as follows:
$p(\text{Anne}) = p(\text{Bob}) = p(\text{Doug}) = p(\text{John}) = ¼$
$H(M) = \Sigma\, p(M) \log_2 1/p(M) = 4 \times ¼ \times \log_2 4 = 2$ bits

If the sex of the recipient were revealed, the conditional probabilities and the equivocation are as follows:
$p(\text{male}) = 1/2, p(\text{female}) = 1/2$
$p_{\text{male}}(\text{Anne}) = 0, p_{\text{male}}(\text{Bob}) = p_{\text{male}}(\text{Doug}) = p_{\text{male}}(\text{John}) = 1/3$
$p_{\text{female}}(\text{Anne}) = 1, p_{\text{female}}(\text{Bob}) = p_{\text{female}}(\text{Doug}) = p_{\text{female}}(\text{John}) = 0$
$H_S(M) = \Sigma_S\, p(S)\ \Sigma_M\, p_S(M) \log_2 1/p_S(M) = 1/2 \times (1\times \log_2 1) + 1/2 \times (3 \times 1/3 \times \log_2 3) = 0 + ½ \times \log_2 3 = ½ \log(3)$ bits

13. Let M be a 6-digit number in a range $[0, 10^6-1]$ enciphered with Caesar type shifted substitution cipher with key K, $0 \le K \le 9$. For example, if $K = 1$, $M = 123456$ is enciphered as 234567. Compute $H(M)$, $H(C)$, $H(K)$, $H_C(M)$ and $H_C(K)$, assuming all values of M and K are equally likely.

Solution:
$p(M) = p(C) = 1/10^6$
$p(K) = 1/10$

$p_C(M) = 1/10$ for the 10 possible messages given ciphertext C
$= 0$ for all other messages

$p_C(K) = 1/10$
(knowing the ciphertext doesn't change the probability of the key)

$H(M) = \Sigma p(M) \log_2 (1/p(M))$
$= 10^6 \times (1/\ 10^6) \times \log_2 10^6 = \log_2 10^6 = 6 \log_2 10$

$H(C) = \Sigma\, p(C) \log_2 (1/p(C))$
$= 10^6 \times (1/\ 10^6) \times \log_2 10^6 = \log_2 10^6 = 6 \log_2 10$

$H(K) = \Sigma\, p(K) \log_2 1/p(K)$
$= 10 \times (1/10) \times \log_2 10 = \log_2 10$

$H_C(M) = \Sigma\, p(C)\, \Sigma\, p_C(M) \log_2 (1/p_C(M))$
$= 10^6 \times (1/\ 10^6) \times (10 \times (1/10) \times \log_2 10 + (10^6 -10) \times 0 \times \log_2 (1/0))$
$= \log_2 10$ (because $\mathit{lim}_{x\to 0}\, x \log_2 (1/x) = 0$)

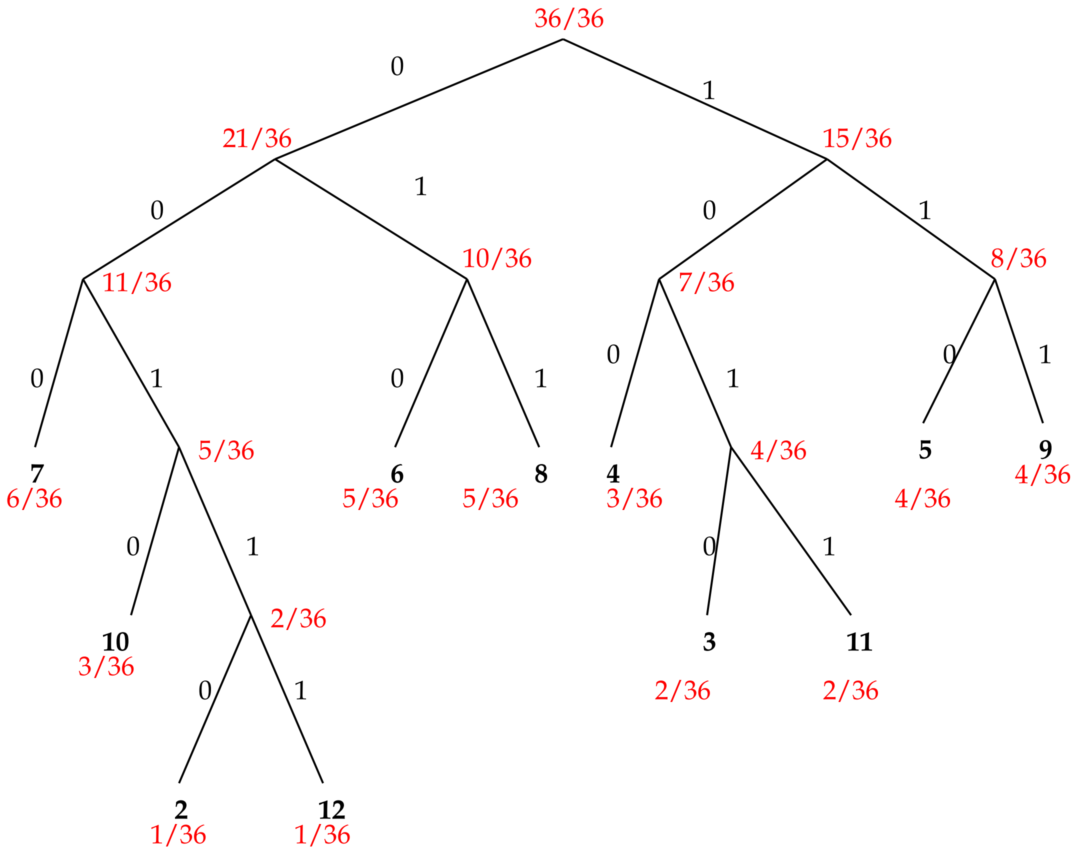
$H_C(K) = \Sigma\, p(C)\, \Sigma\, p_C(K) \log_2 (1/p_C(K))$
$= 10^6 \times (1/\ 10^6) \times (10 \times (1/10) \times \log_2 10) = \log_2 10$

14. Alice rolls two fair dice and records the sum. Bob's task is to ask a sequence of questions with yes/no answers to find out the sum. Help Bob by devising a detailed question strategy that achieves minimum possible ***average*** number of questions.

Solution: When rolling two dice we have 36 possible outcomes, and 11 possible sums from 2 to 12. Need to work out the frequency for each sum and this divided by 36 will give the probability of that sum.

p(2) = 1/36	Outcome (1,1)
p(3) = 2/36	Outcomes (1,2) and (2,1)
p(4) = 3/36	Outcomes (1,3), (2,2) and (3,1)
p(5) = 4/36	Outcomes (1,4), (2,3), (3,2) and (4,1)
p(6) = 5/36	Outcomes (1,5), (2,4), (3,3), (4,2) and (5,1)
p(7) = 6/36	Outcomes (1,6), (2,5), (3,4), (4,3), (5,2) and (6,1)
p(8) = 5/36	Outcomes (2,6), (3,5), (4,4), (5,3) and (6,2)
p(9) = 4/36	Outcomes (3,6), (4,5), (5,4) and (6,3)
p(10) = 3/36	Outcomes (4,6), (5,5) and (6,4)
p(11) = 2/36	Outcomes (5,6) and (6,5)
p(12) = 1/36	Outcome (6,6)

To devise the question strategy we create an optimal encoding.



Now we use this encoding to devise the question strategy that will have the minimum possible average number of questions.

Question1: It the sum any of the following; 3, 4, 5, 9 or 11?

Yes: Go to question 2
No: Go to question 6

Question 2: Is the sum either 5 or 9?

Yes: go to question 3
No: go to question 4

Question 3: Is the sum 9?

Yes: Answer is 9.
No: Answer is 5.

Question 4: Is the sum 4?

Yes: Answer is 4.
No: Go to question 5

Question 5: Is the sum 11?

Yes: Answer is 11.
No: Answer is 3.

Question 6: Is the answer either 6 or 8?

Yes: Go to question 7.
No: Go to question 8.

Question 7: Is the answer 8?

Yes: The answer is 8.
No: The answer is 6.

Question 8: Is the answer 7?

Yes: The answer is 7.
No: Go to question 9.

Question 9: Is the answer 10?

Yes: Answer is 10.
No: Go to question 10.

Question 10: Is the answer 12?

Yes: Answer is 12.
No: Answer is 2.

The average number of questions is 3.306.

15. The accuracy of a certain radio station's weather man at predicting rain is given by the following chart.

	Actual rain	Actual no rain
Predicts rain	1/12	1/6
Predicts no rain	1/12	2/3

For example, 1/12 of the time the weatherman predicts rain when in fact it does rain. Notice that the weatherman is correct 3/4 of the time. An uninformed listener observes that he could be correct 5/6 of the time by simply always predicting no rain. He applies for the weatherman's job. However the station manager declines to hire the listener. Why? Explain using the equivocation of the actual weather condition given the prediction by the weather man, and by the listener.

Solution:

Notation:
Variables: P – prediction; W –weather
p(R) is the probability that it will rain
p(NR) is the probability that it will not rain
$p_R(R)$ is the probability that it will rain, given the prediction that is will rain.
$p_{NR}(R)$ is the probability that it will rain, given the prediction that is will not rain.
$p_R(NR)$ is the probability that it will not rain, given the prediction that is will rain.
$p_{NR}(NR)$ is the probability that it will not rain, given the prediction that is will not rain.

Entropy of the weather when no prediction is known:
p(R) = 1/6
p(NR) = 5/6

$$
\begin{aligned}
H(W) &= \Sigma\, p(W) \log_2 (1/p(W)) \\
&= (1/6) \times \log_2 6 + (5/6) \times \log_2 (6/5) \\
&= (1/6) \times \log_2 3 + (1/6) + (5/6) \times \log_2 3 + (5/6) - (5/6) \times \log_2 5 \\
&= 1 + \log_2 3 - (5/6) \times \log_2 5 \\
&= 0.650 \text{ Bits}
\end{aligned}
$$

The equivocation of the weather when the prediction is known:
a) For the listener.

The probability that the listener predicts rain is 0, and the probability that they predict no rain is 1. The conditional probabilities are as follows.

$p_R(R)$	0
$p_R(NR)$	0
$p_{NR}(R)$	1/6
$p_{NR}(NR)$	5/6

$H_P(W) = \Sigma\, p(P)\, \Sigma\, p_P(W)\ \log_2 (1/p_P(W))$

$= 1 \times ((1/6) \times \log_2 6 + (5/6) \times \log_2 6/5) + 0$

$= (1/6) \times \log_2 3 + (1/6) + (5/6) \times \log_2 3 + (5/6) – (5/6) \times \log_2 5$

$= 1 + \log_2 3 – (5/6) \times \log_2 5$

$= 0.650$ Bits

Thus knowing the prediction by the listener does not lower the uncertainty about the weather.

b) For the weatherman

The probability that the weatherman predicts rain is 1/4, and the probability that they predict no rain is 3/4. The conditional probabilities are as follows.

$p_R(R)$	(1/12) / ((1/12) + (1/6)) = 1/3
$p_R(NR)$	(1/6) / ((1/12) + (1/6)) = 2/3
$p_{NR}(R)$	(1/12) / ((1/12) + (2/3)) = 1/9
$p_{NR}(NR)$	(2/3) / ((1/12) + (2/3)) = 8/9

$H_P(W) = \Sigma\, p(P)\, \Sigma\, p_P(W)\ \log_2 1/p_P(W)$

$= (1/4) \times ((1/3) \times \log_2 3 + (2/3) \times \log_2 (3/2)) + (3/4) \times ((1/9) \times \log_2 9 + (8/9) \times \log_2 (9/8))$

$= (1/4) \times ((1/3) \times \log_2 3 + (2/3) \times \log_2 3 – (2/3)) + (3/4) \times ((2/9) \times \log_2 3 + (16/9) \times \log_2 3 – (24/9))$

$= (1/4) \times \log_2 3 – (1/6) + (3/2) \times \log_2 3 – 2$

$= (7/4) \times \log_2 3 – (13/6)$

$= 0.607$ Bits.

The equivocation in this case represents the uncertainty of the weather, knowing the prediction. So since the weatherman has a lower equivocation, there is less uncertainty about the weather. He does a better job and should be hired over the novice listener.