Number Representation

Qcamp 2018, Experimental Session 01/06/2018

1 Introduction

Philosophically speaking, what is a number? Conventionally we use ten unique symbols such as 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 to represent a number (for eg. like our favourite number 42). Using these 10 symbols to represent a number is called the decimal representation. There are many other ways to represent a number, just like how one can have many names just in different languages.

The aim of the exercise is to gain an appreciation for the following statements:

1. *"There are 10 types of people in this world; those who know binary and those who don't."*
2. *"In some context, dead beef is equivalent to 3735928559."*

1.1 Overview of Number Representation

As a starter, we shall start with a simple question, why do we normally write "ten" as 10? This is because the conventional numeral system we use daily is called the decimal representation (i.e. using only ten symbols to represent our number), and if we start from zero (0), by the time we reached nine (9), we would have ran out of symbols to write. So we just write the '1' symbol in the tens' place and start from the symbol '0' in the ones' place again. This tells us that we have one of the tens' place (10^1) and none in the ones' place (10^0). Hence, we can decompose the number ten into $1 \times 10^1 + 0 \times 10^0$. The position of the symbol has a certain **weight**, so for example the symbol in the tens' place has a weight of ten and the symbol in the hundreds' place to have a weight of hundred so on and so forth. Another point to note is that the number of unique symbols we use to represent numbers under a certain representation is called the **base or radix** of the representation.

For the purposes of today, we will be using the binary representation (radix = 2) and hexadecimal representation (radix = 16). **Numbers represented in binary is prefixed by "0b" and numbers represented in hexadecimal is prefixed by "0x". If not stated explicitly by the conventions above, the number representation is decimal (radix = 10).**

Binary representation is convenient for computers because they are just on/off switches in some sense. In fact, this is why it is a lie if someone tells you, you can only count up to 10 with your 2 hands (assuming you have 10 fingers. If you have 4 fingers only, then you probably serve nice chicken wings! [1]) On the other hand, hexadecimal allows quick reading of binary strings (which we will tell you how later in this worksheet).

[1] This statement is not sponsored by any fast food outlet in any way.

1.2 Converting to Decimal (As Easy As Doing Multiplication!)

To put into practice, if we have a string of symbols how do we translate it into the decimal representation? We just apply the following:

$$\text{Number} = \sum_{n} (\text{symbol value}) \times (\text{radix})^{\text{n}} \tag{1}$$

where n is the position of the symbol from the rightt and we start counting from 0.

The symbols used in each number system in order of increasing value is as follows:

1. Decimal: 0 ,1, 2, 3, 4, 5, 6, 7, 8, 9
2. Binary: 0 ,1
3. Hexadecimal: 0 ,1, 2, 3, 4, 5, 6, 7, 8, 9, a, b, c, d, e, f

By applying Equation 1, we can convert easily from a number basis to decimal system. For eg., starting from the left of the binary representation of the number $0\text{b}1010$:

$$0\text{b}\mathbf{1110} = (\mathbf{1} \times 2^3) + (\mathbf{1} \times 2^2) + (\mathbf{1} \times 2^1) + (\mathbf{0} \times 2^0) = 14$$

Or the hexadecimal representation of the number $0\text{x}2\text{face}$:

$$\begin{aligned}
0\text{x}\mathbf{2face} =& (0\text{x}\mathbf{2} \times 16^4) + (0\text{x}\mathbf{f} \times 16^3) + (0\text{x}\mathbf{a} \times 16^2) + (0\text{x}\mathbf{c} \times 16^1) + (0\text{x}\mathbf{e} \times 16^0) \\
=& (\mathbf{2} \times 16^4) + (\mathbf{15} \times 16^3) + (\mathbf{10} \times 16^2) + (\mathbf{12} \times 16^1) + (\mathbf{14} \times 16^0) \\
=& (\mathbf{2} \times 65536) + (\mathbf{15} \times 4096) + (\mathbf{10} \times 256) + (\mathbf{12} \times 16) + (\mathbf{14} \times 1) \\
=& 195278
\end{aligned}$$

1.3 Conversion between Binary and Hexadecimal (As Easy As Cutting Strings!)

First observation that I would like you to make would be this: 16 is simply 2^4. This means that each hex symbol has a corresponding 4 bits(symbol) long binary representation. This is summarised in the table below:

0	1	2	3	4	5	6	7
0000	0001	0010	0011	0100	0101	0110	0111
8	9	A	B	C	D	E	F
1000	1001	1010	1011	1100	1101	1110	1111

Table 1: Hexadecimal Symbol and Corresponding 4-bit Binary Representation

With this in mind, this means that a number represented by a string of n hexadecimal symbols is represented by a string of $4n$ bit (if we include leading zeroes of course) and vice-versa. Let's look at converting $0\text{x}2\text{face}$ to its binary representation:

$$0\text{x}2\text{face} : (0010)(1111)(1010)(1011)(1110) = 0\text{b}00101111101010111110$$

Converting from binary to hexadecimal is easy as well! Just cut the binary strings up into 4-bit chunks starting from the right and add 0's to the start when necessary[2]. For example, let's convert 0b011101111101101 to hexadecimal:

$$0b11101111101101 : (00)11\ \ 1011\ \ 1110\ \ 1101 = 0x3BED$$

1.4 Conversion from Decimal to a Arbitrary Representation (Divide and Conquer!)

To convert from decimal representation to another is not as straight forward as the other 2 conversions above is not as straightforward, but can be easily done still. All you need to do is by the following steps in the Euclidean algorithm for finding the greatest common divisor[3]:

1. Divide the number you are converting (q_0) by the new representation's radix (b).
2. Get the quotient q_1 and the remainder r_0. ($q_0/b = q_1 * b + r_0$)
3. Divide q_1 by b to obtain $r_1, \ldots q_2$ and repeat until $q_i = 0$. (Essentially you repeat Steps 1-3)
4. String all your r's together, from r_i to r_0 in order to form the answer ($r_{i+1}r_i \ldots r_1r_0$).

Let's convert 47802 to its hexadecimal (radix = 16) representation:

$$\begin{aligned} 47802 =& 2987 \times 16 + \mathbf{10} \\ =& ((186 \times 16 + \mathbf{11}) \times 16) + \mathbf{10} \\ =& (((11 \times 16 + \mathbf{10}) \times 16 + \mathbf{11}) \times 16) + \mathbf{10} \\ =& ((((0 \times 16 + \mathbf{11}) \times 16 + \mathbf{10}) \times 16 + \mathbf{11}) \times 16) + \mathbf{10} \end{aligned}$$

The bold numbers are what we are interested in, hence $47802 = 0\text{xBABA}$.

2 Assignment [10 pts]

Task 1a [1 pt] So by right, what number (in decimal representation) can you count up to using your 2 hands (assuming you have 10 fingers) and if you use a binary number representation?

Task 1b [1 pt] Stretch both your hands out with your open palms facing away from you, fingers facing up. Put down your ring, middle and index fingers on both hands. Assuming a finger that is up represents 1 and a finger that is down represents 0, and the lowest weighted position is on your right little finger (2^0, what number (in decimal representation) are you representing with your hands?

[2] This is also commonly termed as padding.

[3] Refer to this if unclear about the algorithm https://en.wikipedia.org/wiki/Euclidean_algorithm

Task 2 [1 pt] What is 0b1111111011101101 in hexadecimal? Do your working on this paper to show how you arrived at the answer.

Task 3 [2 pt] Convert the number 48879 to its binary representation and hexadecimal. (Hint: The order in which you attempt this question makes life easier.)

Task 4 [2 pts] Convert 0xDEAD to its binary representation and decimal representation. Show your working for the conversion to decimal representation.

Task 5 [3 pts] Osne day you were given the task of creating questions on number representation in a handout that would be given to students during a workshop on experimental quantum physics. You ran out of ideas so you took out a pack of poker cards and was inspired to create the following number representation of radix 13, with the symbols in order of increasing value: A, 2, 3, 4, 5, 6, 7, 8, 9, 10, J, Q, K. You have decided to call this the "poker number representation" and is prefixed by "0p" [4]
To confuse the students even further, 0p2 does not correspond to the decimal value of 2, but corresponds to the decimal value of 1. More examples include: 0pA in decimal is 0, 0pQ in decimal is 11, and0p10 is actually one symbol representing the decimal value of 9.

Convert the Royal Flush hand 0p10JQKA to its **hexadecimal representation**. (Show your working so we can award partial credit if the final answer is not correct)

[4] At the point of writing, we are not sure whether such a basis actually exist or not, and whether or not it has already been named something else or given a convention.