Converting Real Numbers between Bases

From lecture slides CO2103 Chapter 03 on Background, we can see that converting from a number (*N*) in any base (*b*) into decimal can be generalized into (*Expansion Method*):

Given the number
N_b = ... a_n ... $a_3a_2a_1$ a_0,
its decimal equivalent will be:
N_{10} = ... + $a_n \times b_n$ + ... + $a_3 \times b_3$ + $a_2 \times b_2$ + $a_1 \times b_1$ + $a_0 \times b_0$

This can be visualized from the following diagram, where a_3 has a value of $a_3 \times b_3$ for example.

Position	...	n	...	3	2	1	0
Value	...	b^n	...	b^3	b^2	b^1	b^0
Digit	...	a_n	...	a_3	a_2	a_1	a_0

The method described above deals with integers, i.e. whole number. How do we deal with real numbers? To deal with a number with "decimal" point, i.e. with fractions, e.g. 22.34, we need to know how to deal with the fraction, i.e. the 0.34 in the quoted example (22.34). For the number before the "decimal" point, i.e. the 22 in the quoted example (22.34), it can be dealt with as an integer.

The following material was taken from:
http://www.mathpath.org/concepts/Num/frac.htm

Fractions in base *b*: From base *b* to decimal

It is sufficient to consider fractions having values less than 1. Just as an integer in base *b* is represented in terms of powers of *b*, a fraction *N* in base *b* is represented in terms of the negative powers of *b*:

$N = c_1b^{-1} + c_2b^{-2} + c_3b^{-3 + \cdots +} c_nb^{-n} + \ldots$ .... Equation (1)
where $0 \le c_i < b$. This is what is meant when we write *N* as $0.c_1c_2c_3c_4\ldots$

[*Mathematicians annotate the digits starting from most significance position*]

The period in the fraction is called a *radix* point - it is the decimal point for decimal fractions. Converting a fraction given in any base to a corresponding fraction in base 10 is easy.

For instance,

$$0.011_2 = 0 \times 2^{-1} + 1 \times 2^{-2} + 1 \times 2^{-3} = 0 + 1/4 + 1/8 = 0 + 0.25 + 0.125 = 0.375_{10}.$$

This is often called the *Expansion* Method.

There is another way also to do this. That is to write the binary fraction as a ratio. For instance $0.011_2 = (11/1000)_2 = 11_2/1000_2 = (3/8)_{10} = 0.375_{10}$.

Fractions in base *b*: From decimal to base *b*

Converting a decimal fraction to a binary fraction or a fraction in another base, requires more steps. There are three methods: *Repeated Multiplication*, *Repeated Subtraction*, and the *Ratio Method*. The first two methods are, respectively, analogous to the Repeated Division and Repeated Subtraction algorithms for converting integers between different bases.

Repeated Multiplication

How do we convert 0.375_{10} to base 2 or any other base?
To convert, we need to pick up the coefficients $c_1, c_2, c_3, \ldots$

Consider Equation (1). Multiply by *b*.
The integer part of the result is the coefficient c_1.
If the remaining part is zero, we are finished.
If the remaining part is not zero, multiply that by *b*.
The integer part of the result is c_2. If the remaining part is zero, stop; else continue.

Here is why it works.
Consider Equation (1).
$\mathrm{b}N = c_1 + q_1$, where $q_1 = c_2 b^{-1} + c_3 b^{-2} + c_4 b^{-3} + \ldots$
$bq_1 = c_2 + q_2$
… …
$bq_{n-1} = c_n + q_n$,
where we continue until $q_n = 0$.

Now suppose we are given a fraction M_{10} in base 10 and we wish to convert it to a fraction N in base b.

Then, since $N = M_{10}$,
$\mathrm{b}M = \mathrm{b}M_{10}$.

This means, although N is what we are trying to find, since $\mathrm{b}N = \mathrm{b}M_{10}$, we can compute the first step of the algorithm, namely, $\mathrm{b}M_{10} = \mathrm{b}N = \mathrm{c}_1 + \mathrm{q}_1$. That is, we can pick up c_1 as the integer part of $\mathrm{b}M_{10}$, the fractional part of $\mathrm{b}M_{10}$ being q_1. We continue by multiplying the fractional part of $\mathrm{b}M_{10}$ with b, and so on.

Example:

Convert 0.375_{10} to base 2 and base 5.

$2N = 2\times0.375 = 0.75 = 0 + 0.75 \rightarrow\rightarrow c_1 = 0$ and $q_1 = 0.75$
$2q_1 = 1.5 = 1 + 0.5 \rightarrow\rightarrow c_2 = 1$ and $q_2 = 0.5$
$2q_2 = 1 = 1 + 0 \rightarrow\rightarrow c_3 = 1$ and $q_3 = 0$.
When a q-value is 0, we STOP.
So, $0.375_{10} = (0.c_1c_2c_3)_2 = 0.011_2$.

We proceed in the same manner if the decimal fraction is expressed as a ratio.

Thus for 0.375 = 3/8
(3/8)×2 = 6/8 = 0 + 6/8, so the integer part is 0. Peel it off, the fraction remaining is 6/8.
(6/8)×2 = 12/8 = 1 + 4/8, so the integer part is 1. The fraction remaining is 4/8.
(4/8)×2 = 9/8 = 1 + 0. Take these integer parts and write them from the radix point in the order we obtained them.
So, again, $(3/8)_{10} = 0.011_2 = (11/1000)_2$.

Let us convert 0.375_{10} to base 5.

$N = 0.375$
$5N = 5 \times 0.375 = 1.875 = 1 + 0.875 \to\to c_1 = 1$ and $q_1 = 0.875$
$5q_1 = 4.375 = 4 + 0.375 \to\to c_2 = 4$ and $q_2 = 0.375$
Since $q_2 = N$, the computation repeats without ever stopping, and we get
$0.375_{10} = (0.c_1c_2c_1c_2)_5 = (0.1414\ldots)_5$.

Notice that multiplication by the base *b* shifts the radix point (of the fraction in base *b*) to the right, exposing the integer part on the left of the radix point.

Having understood the principle behind the algorithm, we can now simply compute the binary version of the fraction 0.375_{10} as follows:

$$\begin{array}{lrcl} & 0.375 & & \\ \times & 2 & & \\ \cline{1-2} & 0.750 & \Rightarrow & 0 \\ \times & 2 & & \\ \cline{1-2} & 1.500 & \Rightarrow & 1 \\ \times & 2 & & \\ \cline{1-2} & 1.000 & \Rightarrow & 1 \end{array}$$

Repeated Subtraction

Let us proceed via an example by converting 0.375_{10} to binary. Consider the negative powers of 2.

$2^{-1} = 0.5$
$2^{-2} = 1/4 = 0.25$
$2^{-3} = 1/8 = 0.125$
… …

Determine the largest of these fractions that does not exceed 0.375.
Since 0.25 < 0.375 < 0.5, the fraction we need is 0.25.
Then 0.375 = 0.25 + 0.125
0.125 = 0.125 +0
So $0.375 = 1\times2^{-2} + 1\times2^{-3} = 0\times2^{-1} + 1\times2^{-2} + 1\times2^{-3} = 0.011_2$.

[This is reverse process to the Expansion method. We used this method, allocation based on position value, to convert from binary/hexadecimal to decimal]

Example: Convert 0.375_{10} to base 5. Consider the negative powers of 5.

5^{-1} = 1/5 = 0.2
5^{-2} = 1/25 = 0.04
5^{-3} = 1/125 = 0.008
5^{-4} = 1/625 = 0.0016

Determine the largest of these fractions that does not exceed 0.375.
Since 0.375 > 0.2, the fraction we need is 0.2.
Then 0.375 = 0.2 + 0.175
0.175 = 0.04 + 0.135
But 0.135 > 0.04. So we subtract 0.04 again.
0.135 = 0.04 + 0.095. Subtract 0.04 again.
0.095 = 0.04 + 0.055
0.055 = 0.04 + 0.015
Now 0.04 > 0.015 > 0.008
0.015 = 0.008 + 0.007
0.007 = 4 ×0.0015 + 0.0006
… …
Thus 0.375 = $1 \times 5^{-1} + 4\times 5^{-2} + 1 \times 5^{-3} + 4\times 5^{-4} + \ldots$

Clearly, the Repeated Subtraction method for fractions is easy to understand but messy to carry out. Therefore, we shall omit giving the (simple) theory of this method.

The Ratio Method

This is another way to convert a decimal fraction to binary. Here we first express the fraction as a ratio. Thus $0.375_{10} = (375/1000)_{10} = (111111111/1111011000)_2$ and then do the division in binary.

Comparison of the three methods

It is clear that *Repeated Multiplication* is the easiest at least for computing by hand. The R*atio Method* would be easy except for the division that has to be carried out in the new base. The *Repeated Subtraction Method* distinguished base 2 in that only one subtraction was needed in each step of obtaining a digit of the binary fraction.

Converting a fraction between bases

Now that we know how to convert a fraction from DECIMAL to any other base, how about converting a fraction from any integer base to any other integer base? The case we presented is a special case, namely, from decimal fraction to fraction in any base.

Let us consider converting 0.011_2 to the corresponding fraction in base 5. From the discussion above, we know how to do this by first converting 0.011_2 to decimal and then converting the decimal to base 5. But this goes via the decimal. A direct conversion is to multiply 0.011_2 with 5 expressed in base 2. We get $0.011_2 \times 101_2 = 1.111_2$. The integer part is 1. This is the coefficient c_1 for our fraction in base 5. CONTINUE WITH THE NON-INTEGER PART AND MULTIPLY IT BY 5 EXPRESSED IN BASE 2: $0.111_2 \times 101_2 = 100.011_2$. The integer part is 4 in base 5. This is the coefficient c_2 for our fraction in base 5. Continue. But the remaining non-integer part is 0.011_2 which is what we started with; so the coefficients repeat; and we get $0.011_2 = (0.1414\ldots)_5$.

Exercises

1. Convert the following decimal numbers into their equivalent binary numbers:
   a) 0.3125
   b) 12.125
   c) 2.345
   d) 0.123
   (Ans. a) 0.0101, b) 1100.001, c) 10.010**11000010100011110101**...,
   d) 0.0001111101111100111011011001...)

2. Determine the decimal equivalent of the following binary numbers:
   a) 1011.11
   b) 0.01001
   c) 1110.0111
   d) 0.110101
   (Ans. a) 11.75, b) 0.28125, c) 14.4375, d) 0.828125)