CM 1015 Computational Mathematics

Kevin Juandi
email: kjuandi@gmail.com

April 21, 2023

Preface

I wrote this note after finishing the course, so the content might not reflect the current version of the course. It is mostly based on my handwritten personal notes. I personally feel this course should be called "Foundation Mathematics" instead of "Computational Mathematics" because of the lack of "Numerical Methods" and probably some other things people more familiar with the topic would say. If you spot any error please don't hesitate to contact me via slack or mail me. I will update the notes along with the pace of the course, every Saturday or Sunday before the start of the week. If you need help with the subject, don't hesitate to contact me.

Contents

Chapter 1

Number Bases, Conversion and Operations

Reading Materials:
Croft, A. and R. Davidson *Foundation maths.* (Harlow: Pearson, 2016) 6th edition. **Chapter 14 Number Bases**

1.1 Number Bases

Decimal System

The numbers that we commonly used are based on 10.
For example:

$$\begin{aligned} 253 &= 200 + 50 + 3 \\ &= 2(100) + 5(10) + 3 \\ &= 2(10^2) + 5(10^1) + 3(10^0) \end{aligned} \tag{1.1}$$

Binary System

A binary system uses base 2, it only consist of 2 digits, 0 and 1.
Numbers in base 2 are called binary digits or simply bits.
Consider the binary number 110101_2. As the base is 2, this means that power of 2 essentially replace powers of 10. Let us convert it to base 10.

$$\begin{aligned} 110101_2 &= 1(2^5) + 1(2^4) + 0(2^3) + 1(2^2) + 0(2^1) + 1(2^0) \\ &= 1(32) + 1(16) + 0(8) + 1(4) + 0(2) + 1(1) \\ &= 32 + 16 + 4 + 1 \\ &= 53_{10} \end{aligned} \tag{1.2}$$

Octal System

Octal numbers use 8 as a base. The eight digits used in the octal system are 0, 1, 2, 3, 4, 5, 6 and 7. Octal numbers use powers of 8, just as decimal numbers use powers of 10 and binary numbers use powers of 2. Example:

$$\begin{aligned} 325_8 &= 3(8^2) + 2(8^1) + 5(8^0) \\ &= 3(64) + 2(8) + 5(1) \\ &= 192 + 16 + 5 \\ &= 213_{10} \end{aligned} \tag{1.3}$$

Hexadecimal System

Hexadecimal system use 16 as a base. The digits are 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E and F. Example:

$$\begin{aligned} 93\mathrm{A}_{16} &= 9(16^2) + 3(16^1) + \mathrm{A}(16^0) \\ &= 9(256) + 3(16) + 10(1) \\ &= 2304 + 48 + 10 \\ &= 2362_{10} \end{aligned} \tag{1.4}$$

1.2 Number Conversion

1.2.1 Converting from Decimal to Other Number Base

The no-brainer way is to divide the number by the base, the remainder would be the last digit of the new number base. Keep dividing the quotient until it is smaller than the number base. Let us convert 253_{10} as example.

$$\begin{aligned} 2\overline{)253} &= 126 \text{ with remainder } 1 \\ 2\overline{)126} &= 63 \text{ with remainder } 0 \\ 2\overline{)63} &= 31 \text{ with remainder } 1 \\ 2\overline{)31} &= 15 \text{ with remainder } 1 \\ 2\overline{)15} &= 7 \text{ with remainder } 1 \\ 2\overline{)7} &= 3 \text{ with remainder } 1 \\ 2\overline{)3} &= 1 \text{ with remainder } 1 \\ 2\overline{)1} &= 0 \text{ with remainder } 1 \end{aligned} \tag{1.5}$$

Thus, 253_{10} is 11111101_2 in binary. We could do the same to other number bases.

Another method is by listing the powers of the base, compare and subtract. Using the same number as example.

$$\begin{array}{ccc} 2^0 = 1 & 2^1 = 2 & 2^2 = 4 \\ 2^3 = 8 & 2^4 = 16 & 2^5 = 32 \\ 2^6 = 64 & 2^7 = 128 & 2^8 = 256 \end{array}$$

From here we compare the number that we are going to convert with the list:

$$\begin{aligned} 253 - 1(128) &= 125 \\ 125 - 1(64) &= 61 \\ 61 - 1(32) &= 29 \\ 29 - 1(16) &= 13 \\ 13 - 1(8) &= 5 \\ 5 - 1(4) &= 1 \\ 1 - 0(2) &= 1 \\ 1 - 1(1) &= 0 \end{aligned} \tag{1.6}$$

Thus, 253_{10} is 11111101_2 in binary. Like the other method, we can also do this to convert to other number bases.

1.2.2 Conversion with Binary Number

Converting binary numbers to Octal or Hexadecimal and vice versa is very straightforward. It can be performed without converting to Decimal first. Let us use number 11100110_2 as example:

$$\begin{aligned} \underbrace{11}_{3}\underbrace{100}_{4}\underbrace{110}_{6} &= 346_8 \\ \underbrace{1110}_{\mathrm{E}}\underbrace{0110}_{6} &= \mathrm{E}6_{16} \end{aligned} \tag{1.7}$$

1.2.3 Non-integer Number Conversion

Converting non-integer number might look counterintuitive and intimidating. It is actually rather simple. Let us convert 17.375_{10} to binary.

$$\begin{aligned} 17.375 &= 10 + 7 + 0.3 + 0.07 + 0.005 \\ &= 1(10^1) + 7(10^0) + 3(10^{-1}) + 7(10^{-2}) + 5(10^{-3}) \end{aligned}$$

Converting to binary, $17_{10} = 10001_2$. But how about the decimal point? We multiply them by two until we are left with whole number

$$\begin{aligned} 0.375 \times 2 = 0.75 = 0 + 0.75 &\text{ we have 0 at power } -1 \\ 0.75 \times 2 = 1.5 = 1 + 0.5 &\text{ we have 1 at power } -2 \\ 0.5 \times 2 = 1.0 = 1 &\text{ we have 1 at power } -3 \end{aligned} \tag{1.8}$$

Thus, $17.375_{10} = 10001.011_2$
The reverse is much simpler.

$$\begin{aligned}
1101.101_2 &= 1(2^3) + 1(2^2) + 0(2^1) + 1(2^0) + 1(2^{-1}) + 0(2^{-2}) + 1(2^{-3}) \\
&= 1(8) + 1(4) + 0(2) + 1(1) + 1(0.5) + 0(0.25) + 1(0.125) \\
&= 8 + 4 + 1 + 0.5 + 0.125 \\
&= 13.625_{10}
\end{aligned} \tag{1.9}$$

1.3 Operations with Binary Number

1.4 Common Pitfalls

Bibliography