

# 第二单元学习笔记

yinxuhao [xuhao\_yin@163.com]

December 16, 2022

## Contents

<b>1</b>	<b>引言</b>	<b>2</b>
<b>2</b>	<b>信息存储</b>	<b>2</b>
2.1	十六进制表示法 . . . . .	2
2.2	字数据大小 . . . . .	3

信息的表示和处理

1 引言

孤立地讲，单个的位不是非常有用，将位组合在一起，再加上某种解释 (interpretation)，即赋予不同的可能位模式以含意。我们就能表示任何有限集合的元素。

三种重要的数字表示：

- 1. 无符号unsigned编码给予传统的二进制表示法
- 2. 补码two's-complement编码是表示有符号整数的最常见的方式。
- 3. 浮点数floating-point编码是表示实数的科学计数法的以 2 为基数的版本。数据溢出overflow是产生 bug 的一大原因。负数下溢产生极大的正数；正数上溢产生极小的负数。

浮点运算有完全不同的数学属性。

- 1. 由于表示的精度有限，浮点运算是不可结合的。例如

(3.14 + 1e20) - 1e20 = 0.0

but

(3.14 + 1e20 - 1e20) = 3.14

- 2. 该属性不同的原因，是处理数字表示有限性的方式不同——  
整数虽只能编码一个相对较小的数值范围，然该表示法是精确的；  
浮点数虽可以编码宇哥怀大的数值范围，但这种表示只是近似的。  
书中建议的本章学习方式：深入[学习数学语言](#)，编写公式和方程式，以及重要属性的推导。

2 信息存储

大多数计算机使用 8 位的块或者字节作为最小的可寻址内存单位，而不是内存中单独的比特。

机器级程序将内存视为一个非常大的字节数组，称为虚拟内存，所有可能的地址的集合称为虚拟地址空间virtual address space.

每个程序对象可以简单地视为一个字节块，而程序本身就是一个字节序列。

2.1 十六进制表示法

Hex digit	0	1	2	3	4	5	6	7
Decimal value	0	1	2	3	4	5	6	7
Binary value	0000	0001	0010	0011	0100	0101	0110	0111
Hex digit	8	9	A	B	C	D	E	F
Decimal value	8	9	10	11	12	13	14	15
Binary value	1000	1001	1010	1011	1100	1101	1110	1111

Figure 1: 十六进制表示法。每个十六进制数字都对 16 个值中的一个进行了编码

十六进制转二进制：将十六进制的每一位转换为二进制格式，然后拼接。例如：

十六进制	1	7	3	A	4	C
二进制	0001	0111	0011	1010	0100	1100

所以  $binary_{0x173a4c_{16}} = 000101110011101001001100_2$ 。

二进制转十六进制：将二进制从右到左做 4 个一组的划分，如最左侧不足 4 位则以 0 补之。然后将每个 4 位转换为对应的十六进制数字拼接即可。例如：

二进制	11	1100	1010	1101	1011	0011
十六进制	3	C	A	D	B	3

所以， $hex_{111100101010110110011_2} = 3cadb3_{16}$

## 2.2 字数据大小

每台计算机都有一个字长，指明指针数据的标称大小。

C 数据类型的典型大小见下图：

C declaration		Bytes	
Signed	Unsigned	32-bit	64-bit
[signed] char	unsigned char	1	1
short	unsigned short	2	2
int	unsigned	4	4
long	unsigned long	4	8
int32_t	uint32_t	4	4
int64_t	uint64_t	8	8
char *		4	8
float		4	4
double		8	8

Figure 2: 基本 C 数据类型的典型大小 (以字节为单位)