Computer Science Illuminated
SECOND EDITION
Nell Dale • John Lewis

Chapter 02

Number System

从这里开始

Communication
Application
Operating System
Programming
Hardware
Information

2-2

“Information” definition by wiki

Information is that which informs, as well as that from which knowledge and data can be derived. As it regards data, the information's existence is not necessarily coupled to an observer , while in the case of knowledge, the information requires a cognitive observer.

For example:

observe → **Apple Red Ill**

观察 → 苹果 玫瑰红 叁

As a property in physics
As sensory input
As an influence which leads to a transformation
As representation and complexity

2-3

大道至简

2-4

二进制（Binary）

穿孔卡

八卦

二进制

2-5

Chapter Goals

- **Know the different types of numbers**
- **Describe positional notation**
- **Convert numbers in other bases to base 10**
- **Convert base 10 numbers into numbers of other bases**
- **Describe the relationship between bases 2, 8, and 16**
- **Explain computing and bases that are powers of 2**

2-6

Numbers

Natural Numbers
Zero and any number obtained by repeatedly adding one to it.

Examples: 100, 0, 45645, 32

Negative Numbers
A value less than 0, with a – sign

Examples: -24, -1, -45645, -32

2-7

Numbers

Integers
A natural number, a negative number, zero

Examples: 249, 0, - 45645, - 32

Rational Numbers
An integer or the quotient of two integers

Examples: -249, -1, 0, 3/7, -2/5

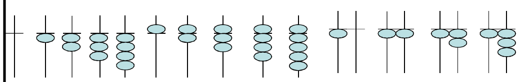
2-8

Counting(计数)

0 1 2 3 4 5 6 7 8 9 10 11 12 13

I II III IV V VI VII VIII IX X XI XII XIII

甲子 乙丑 丙寅 丁卯 戊辰 己巳 庚午 辛未 壬申 癸酉 甲戌 乙亥 丙子



2-9

Natural Numbers

How many ones are there in "642" ?

600 + 40 + 2 ?
Or is it
384 + 32 + 2 ?
Or maybe…
1536 + 64 + 2 ?

2-10

Natural Numbers

Aha!

642 is 600 + 40 + 2 in **BASE（基） 10**

The **base** of a number determines the number of digits and the value of digit positions

2-11

Positional Notation（进位制/位值计数法）

Continuing with our example…
642 in base 10 *positional notation* is:

6 x 10^2 = 6 x 100 = 600
+ 4 x 10^1 = 4 x 10 = 40
+ 2 x 10^0 = 2 x 1 = 2 = 642 in base 10

This number is in base 10

The power indicates the position of the number

2-12

Positional Notation

As a formula:

$d_n * R^{n-1} + d_{n-1} * R^{n-2} + \ldots + d_2 * R + d_1$

R is the base of the number

n is the number of digits in the number

d is the digit in the i^{th} position in the number

$$\mathbb{N} = \sum_{i=1}^{n} d_i R^{i-1}$$

$(642)_{10}$ is $6_3 * 10^2 + 4_2 * 10 + 2_1$

2-13

Positional Notation

What if 642 has the base of 13?

+ 6 x 13^2	= 6 x 169	= 1014
+ 4 x 13^1	= 4 x 13	= 52
+ 2 x 13^0	= 2 x 1	= 2
		= 1068 in base 10

642 in base 13 is equivalent to 1068 in base 10

2-14

Binary Number（二进制）

Decimal is base 10 and has 10 digits:
0,1,2,3,4,5,6,7,8,9

Binary is base 2 and has 2 digits:
0,1

For a number to exist in a given number system, the number system must include those digits. For example, the number 284 only exists in base 9 and higher.

2-15

Bases Higher than 10

How are digits in bases higher than 10 represented?

With distinct symbols for 10 and above.

Base 16 has 16 digits:
0,1,2,3,4,5,6,7,8,9,A,B,C,D,E, and F

2-16

Converting Octal to Decimal

What is the decimal（十进制）equivalent of the octal（八进制）number 642?

6 x 8^2	= 6 x 64	= 384
+ 4 x 8^1	= 4 x 8	= 32
+ 2 x 8^0	= 2 x 1	= 2
		= 418 in base 10

2-17

Converting Hexadecimal to Decimal

What is the decimal equivalent of the hexadecimal（十六进制）number DEF?

D x 16^2	= 13 x 256	= 3328
+ E x 16^1	= 14 x 16	= 224
+ F x 16^0	= 15 x 1	= 15
		= 3567 in base 10

Remember, the digits in base 16 are 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F

2-18

Converting Binary to Decimal

What is the decimal equivalent of the binary number 1101110?

$$\begin{aligned}
&1 \times 2^6 = 1 \times 64 = 64\\
+&1 \times 2^5 = 1 \times 32 = 32\\
+&0 \times 2^4 = 0 \times 16 = 0\\
+&1 \times 2^3 = 1 \times 8 = 8\\
+&1 \times 2^2 = 1 \times 4 = 4\\
+&1 \times 2^1 = 1 \times 2 = 2\\
+&0 \times 2^0 = 0 \times 1 = 0\\
&= 110 \text{ in base } 10
\end{aligned}$$

2-19

Arithmetic in Binary

Remember that there are only 2 digits in binary, 0 and 1

Position is key, carry values are used:

```
  0111111    <- Carry Values
  1010111
 +1001011
 10100010
```

Subtracting Binary Numbers

Remember borrowing? **Apply that concept here:**

```
    12
    202
  1010111
 - 111011
  0011100
```

2-21

Converting Binary to Octal

- Groups of Three (from right)
- Convert each group

10101011   10 101 011
                       2   5   3

10101011 is 253 in base 8

2-22

Converting Binary to Hexadecimal

- Groups of Four (from right)
- Convert each group

10101011   1010 1011
                       A   B

10101011 is AB in base 16

2-23

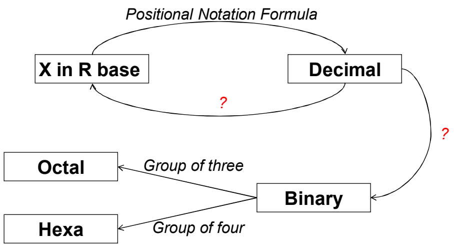
Power of 2 Number System

Binary	Octal	Hexa
0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	4	4
0101	5	5
0110	6	6
0111	7	7

Binary	Octal	Hexa
1000		8
1001		9
1010		A
1011		B
1100		C
1101		D
1110		E
1111		F

2-24

Converting for Natural Numbers



2-25

Converting Decimal to Other Bases

Algorithm for converting base 10 to other bases

While the quotient（商） is ***not*** zero

Divide the decimal number by the new base

Make the remainder（？） the next digit to the left in the answer

Replace the original dividend with the quotient

2-26

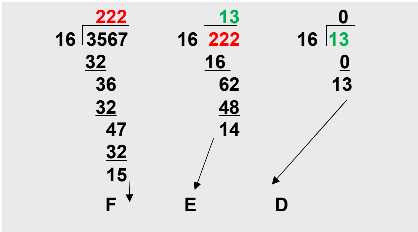
Converting Decimal to Hexadecimal

Try a Conversion

The base 10 number 3567 is what number in base 16?

2-27

Converting Decimal to Hexadecimal



2-28

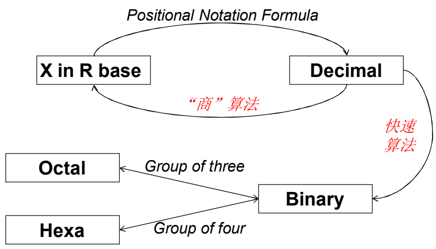
十进制转二进制

1610/2=805……0；
805 /2=402……1；
402 /2=201……0；
201 /2=100……1；
100 /2=50 ……0；
50 /2=25 ……0；
25 /2=12 ……1；
12 /2=6 ……0；
6 /2=3 ……0；
3 /2=1 ……1；
1 /2=0 ……1.

$(1610)_{10}$= $(110\ 0100\ 1010)_2$

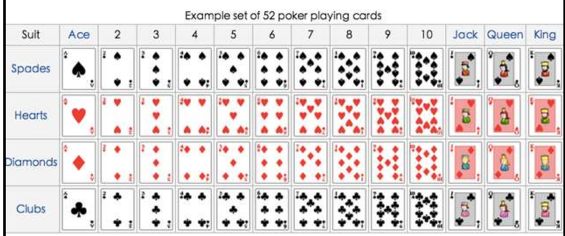
2-29

Converting for Natural Numbers



2-30

What is the No.27 Card?



2-31

作业（part 1 of 2）

- 进制转换
  - ① 1分12秒 = （ ）毫秒
  - ② （7A）$_{13}$ = （ ）$_{10}$
  - ③ （7A）$_{16}$ = （ ）$_{10}$
  - ④ （7A）$_{16}$ = （ ）$_{2}$ = （ ）$_{8}$
  - ⑤ （1011011）$_{2}$ = （ ）$_{8}$ = （ ）$_{16}$
  - ⑥ （678）$_{10}$ = （ ）$_{2}$ = （ ）$_{8}$
  - ⑦ （111）$_{2}$ + （101）$_{2}$ = （ ）$_{2}$

2-32

作业（part 2 of 2）

在浏览器中输入［**http://en.wikipedia.org/**］进入维基百科；**Search**以下关键词，并解释。
1)Information
2)Positional notation
3)Algorithm
4)Software bug

写出以下概念的英文单词：
1)十进制
2)二进制
3)八进制
4)十六进制

2-33

HISTORY: **Bi-quinary coded decimal**

- What is ?
  - Bi-quinary coded decimal is a numeral encoding scheme used in many abacuses and in some early computers.
  - The term bi-quinary indicates that the code comprises both a two-state (bi) and a five-state (quinary) component.

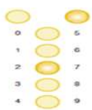
2-34

Bi-quinary in some early computers

- IBM 650(1950s) – 7 bits (two 'bi' bits: 0 5 and five 'quinary' bits: 0 1 2 3 4) with error checking (exactly one 'bi' bit and one 'quinary' bit set in a valid digit)

Value	05-01234 Bits
0	10-10000
1	10-01000
2	10-00100
3	10-00010
4	10-00001
5	01-10000
6	01-01000
7	01-00100
8	01-00010
9	01-00001

IBM 650 front panel



2-35

Bi-quinary in some early computers

- UNIVAC Solid State – 4 bits (one 'bi' bit: 5 and three binary coded 'quinary' bits: 4 2 1) with 1 **parity check bit**

Value	p-5-421 Bits
0	1-0-000
1	0-0-001
2	0-0-010
3	1-0-011
4	0-0-100
5	0-1-000
6	1-1-001
7	1-1-010
8	0-1-011
9	1-1-100

2-36

Bi-quinary & Binary

- The Bi-quinary example illustrates:
  - The bi-quinary difference between abacus and IBM360 and UNIVAC?
  - Why had bi-quinary used in some early computer? Give your explanation
  - Simulation is a good innovation strategy?

2-37