

These slides are being provided with permission from the copyright for in-class (CS2208B) use only. The slides must not be reproduced or provided to anyone outside of the class.

All download copies of the slides and/or lecture recordings are for personal use only. Students must destroy these copies within 30 days after receipt of final course evaluations.

Tutorial 01: Number Systems

Computer Science Department

CS2208: Introduction to Computer Organization and Architecture

Winter 2021-2022

Instructor: Mahmoud R. El-Sakka

Office: MC-419

Email: elsakka@csd.uwo.ca

Phone: 519-661-2111 x86996

Number Systems

- In positional notation number systems
 - Numbers are *represented* (*encoded*) using digits
 - Each digit has a value and a place
 - Each place has a weight
 - for example, in decimal, we have the *ones place*, *tens place*, *hundreds place*, ...

Number Systems

- A radix or base is
 - the number of unique digits, *including zero*, used to represent numbers in a positional numeral system.
- With the use of a *radix point*, the notation can be extended to include *fractions* and *real numbers*.

Number Systems

■ Examples of positional numeral systems

- Decimal is base-10 → {0, 1, 2, 3, 4, 5, 6, 7, 8, and 9}
- Binary is base-2 → {0, and 1}
- Quaternary is base-4 → {0, 1, 2, and 3}
- Octal is base-8 → {0, 1, 2, 3, 4, 5, 6, and 7}
- Hexadecimal is base-16 → {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E and F}

- Trinary is base-3 → {0, 1, and 2}
- Quinary is base-5 → {0, 1, 2, 3, and 4}
- Senary is base-6 → {0, 1, 2, 3, 4, and 5}
- Septenary is base-7 → {0, 1, 2, 3, 4, 5, and 6}
- Nonary is base-9 → {0, 1, 2, 3, 4, 5, 6, 7, and 8}
- Duodecimal is base-12 → {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, and B}
- Sexagesimal is base-60 → {0, 1, 2, 3, 4, 5, ..., 58 and 59}

You need to know how
to convert values from
one system to
the other

Conversion from any other base system to decimal

□ If the original number in base b is:

$$(a_{n-1} a_{n-2} \dots a_i \dots a_1 a_0 \cdot a_{-1} a_{-2} \dots a_{-m})_b$$

□ The *decimal* value of this number is defined as

$$N_{10} = (a_{n-1}b^{n-1} + \dots + a_i b^i + \dots + a_1 b^1 + a_0 b^0 + a_{-1} b^{-1} + a_{-2} b^{-2} + \dots + a_{-m} b^{-m})_{10}$$

Conversion from any other base system to decimal

■ Example 1: Convert $2E8_{16}$ to *decimal*

$$\begin{aligned}
 2E8_{16} &= 2 \times 16^2 + E \times 16^1 + 8 \times 16^0 \\
 &= 2 \times 256 + 14 \times 16 + 8 \times 1 \\
 &= 512 + 224 + 8 \\
 &= 744_{10}
 \end{aligned}$$

This question can be asked
as follow:
Convert the hexadecimal
value $2E8$ to *decimal*

Calculators are not allowed during exams.
You need to improve your mental math skills.

During exams, calculations will be simplified.

Yet, when you answer the assignment/quiz questions, you
may want to use calculators, as simplifying the
calculations are not considered in the assignment/quiz.

10 = A
11 = B
12 = C
13 = D
14 = E
15 = F

Conversion from any other base system to decimal

- Example 2: Convert 361_8 to *decimal*

$$\begin{aligned} 361_8 &= 3 \times 8^2 + 6 \times 8^1 + 1 \times 8^0 \\ &= 3 \times 64 + 6 \times 8 + 1 \times 1 \\ &= 192 + 48 + 1 \\ &= 241_{10} \end{aligned}$$

Conversion from any other base system to decimal

■ Example 3: Convert 0.361_8 to *decimal*

$$\begin{aligned}
 0.361_8 &= 3 \times 8^{-1} + 6 \times 8^{-2} + 1 \times 8^{-3} \\
 &= 3 \times 0.125 + 6 \times 0.015625 + 1 \times 0.001953125 \\
 &= 0.375 + 0.09375 + 0.001953125 \\
 &= 0.470703125_{10}
 \end{aligned}$$

Another method:

$$\begin{aligned}
 0.361_8 &= 361_8 / 1000_8 \\
 &= (3 \times 8^2 + 6 \times 8^1 + 1 \times 8^0) / (1 \times 8^3) \\
 &= (3 \times 64 + 6 \times 8 + 1 \times 1) / (1 \times 512) \\
 &= (192 + 48 + 1) / (512) \\
 &= 241 / 512 \\
 &= 0.470703125_{10}
 \end{aligned}$$

Conversion from any other base system to decimal

■ Example 4: 12.112_3 to *decimal*

$$12.112_3$$

$$= 1 \times 3^1 + 2 \times 3^0 + 1 \times 3^{-1} + 1 \times 3^{-2} + 2 \times 3^{-3}$$

$$= 1 \times 3 + 2 \times 1 + 1 \times 0.33333 + 1 \times 0.11111 + 2 \times 0.03703$$

$$= 3 + 2 + 0.33333 + 0.11111 + 0.07406 = 5.5185_{10}$$

Another method:

$$12.112_3 = 12112_3 / 1000_3$$

$$= (1 \times 3^4 + 2 \times 3^3 + 1 \times 3^2 + 1 \times 3^1 + 2 \times 3^0) / (1 \times 3^3)$$

$$= (81 + 54 + 9 + 3 + 2) / 27$$

$$= 149 / 27 = 5.5185_{10}$$

Conversion from decimal to any other base system

■ Division Method (for integer numbers)

- Initialize the **quotient** by the value of the *decimal number*
- *Repeat*:
 - **Divide** the **quotient** from the previous stage **by the new base** to get
 - A new **quotient** (the whole number)
 - A *remainder*
 - The *remainder* here is *the next least significant digit* in the new number
- Until* the new **quotient** becomes 0.

Conversion from decimal to any other base system

- Example 5: Convert 14_{10} to binary

Binary means the new base is 2

- $14/2 = 7$ Remainder: 0 → This is the least significant binary digit
Quotient = $7 \neq 0$ → continue
- $7/2 = 3$ Remainder: 1 → This is the 2nd least significant binary digit
Quotient = $3 \neq 0$ → continue
- $3/2 = 1$ Remainder: 1 → This is the 3rd least significant binary digit
Quotient = $1 \neq 0$ → continue
- $1/2 = 0$ Remainder: 1 → This is the 4th least significant binary digit
Quotient = 0 → exit the *repeat-until* control structure

□ $14_{10} = 1110_2 \cdot \bullet \bullet \bullet$

Note that, it is 1110_2
It is NOT 0111_2

Conversion from decimal to any other base system

■ Example 6: Convert 2477_{10} to hexadecimal:

Hexadecimal means the new base is 16

□ $2477/16 = 154$ Remainder: 13 → This is the least significant Hex digit

Quotient = $154 \neq 0$ → continue

□ $154/16 = 9$ Remainder: 10 → This is the 2nd least significant Hex digit

Quotient = $9 \neq 0$ → continue

□ $9/16 = 0$ Remainder: 9 → This is the 3rd least significant Hex digit

Quotient = 0 → exit the *repeat-until* control structure

□ $2477_{10} = 9AD_{16}$

Note that, it is $9AD_{16}$
It is NOT $DA9_{16}$

10 = A	13 = D
11 = B	14 = E
12 = C	15 = F

Conversion from decimal to any other base system

■ Multiplication Method (for fraction numbers)

- Initialize the **fraction** by the value of the *fractional decimal number*
 - *Repeat*:
 - **Multiply** the **fraction** from the previous stage **by the new base** to get
 - A *whole number*
 - A new *fraction*
 - The *whole number* here is *the next digit to the right after the radix point* in the new number
- Until* the new **fraction** becomes 0.

This condition might be *reached* or *not*.

If reached,

then we will have 100% accurate result.

If not,

then we have to approximate the result.

Conversion from decimal to any other base system

- Example 7: Convert 0.017578125_{10} to hexadecimal

Hexadecimal means the new base is 16

$$\square 0.01757812 \times 16 = 0.28125$$

whole number: 0 \rightarrow the next digit to the right after the radix point

fraction = $0.28125 \neq 0 \rightarrow$ continue

$$\square 0.28125 \times 16 = 4.5$$

whole number: 4 \rightarrow the next digit to the right after the radix point

fraction = $0.5 \neq 0 \rightarrow$ continue

$$\square 0.5 \times 16 = 8.0$$

whole number: 8 \rightarrow the next digit to the right after the radix point

fraction = $0.0 \rightarrow$ exit the *repeat-until* control structure

$$\square 0.017578125_{10} = 0.048_{16}$$

Conversion from decimal to any other base system

- Example 8: Convert 255.017578125_{10} to hexadecimal:

Hexadecimal means the new base is 16

$$255.017578125_{10} = 255_{10} + 0.017578125_{10}$$

Using the *division method*: $255_{10} \rightarrow FF_{16}$

Using the *multiplication method*: $0.017578125_{10} \rightarrow 0.048_{16}$

$$255.017578125_{10} = FF.048_{16}$$

Conversion from decimal to any other base system

- Example 9: Convert 0.85_{10} to hexadecimal

Hexadecimal means the new base is 16

□ $0.85 \times 16 = 13.6$

whole number: 13 → *the next digit to the right after the radix point*

fraction = $0.6 \neq 0$ → continue

□ $0.6 \times 16 = 9.6$

whole number: 9 → *the next digit to the right after the radix point*

fraction = $0.6 \neq 0$ → continue

□ $0.6 \times 16 = 9.6$

whole number: 9 → *the next digit to the right after the radix point*

fraction = $0.6 \neq 0$ → continue

□ ...

□ $0.85_{10} = 0.D99999...9_{16}$

□ Can be approximated in 4 digits after the radix point, for example, as

- $0.D999_{16}$ (using truncation) or as

- $0.D99A_{16}$ (using rounding)

Conversion between any two bases, other than decimal

- This task can be done in **two steps**:
 - Convert from the **source base** to the **decimal**
 - Convert from the **decimal** to the **destination base**

Conversion between any two bases, other than decimal

- **Example 10**: Convert $2E8_{16}$ to *octal*

$$\begin{aligned} 2E8_{16} &= 2 \times 16^2 + E \times 16^1 + 8 \times 16^0 \\ &= 2 \times 256 + 14 \times 16 + 8 \times 1 \\ &= 512 + 224 + 8 = 744_{10} \end{aligned}$$

10 = A	13 = D
11 = B	14 = E
12 = C	15 = F

$744/8 = 93$ Remainder: 0 → This is the least significant octal digit

Quotient = 93 $\neq 0$ → continue

$93/8 = 11$ Remainder: 5 → This is the 2nd least significant octal digit

Quotient = 11 $\neq 0$ → continue

$11/8 = 1$ Remainder: 3 → This is the 3rd least significant octal digit

Quotient = 1 $\neq 0$ → continue

$1/8 = 0$ Remainder: 1 → This is the 4th least significant octal digit

Quotient = 0 $\neq 0$ → exit the *repeat-until* control structure

$$2E8_{16} = 744_{10} = 1350_8$$

Conversion between any two bases, other than decimal (Special cases)

■ Binary to octal or hexadecimal:

□ *Binary to octal conversion*

- Group bits in three's, starting from the binary point
(pad the last group with 0's, if needed)
- Convert each of these three-bit group into an octal digit

□ *Binary to hexadecimal conversion*

- Group bits in four's, starting from the binary point
(pad the last group with 0's, if needed)
- Convert each of these four-bit group into a hexadecimal digit

Conversion between any two bases, other than decimal (Special cases)

- Example 11: Convert 11001111_2 to *octal*

11001111_2

→ $011\ 001\ 111_2$

→ 317_8

0 = 000

1 = 001

2 = 010

3 = 011

4 = 100

5 = 101

6 = 110

7 = 111

Conversion between any two bases, other than decimal (Special cases)

- Example 12: Convert 1111010101_2 to hexadecimal

1111010101_2

→ $0011\ 1101\ 0101_2$

→ $3D5_{16}$

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

Conversion between any two bases, other than decimal (Special cases)

■ Octal or hexadecimal to binary:

□ *Octal to binary conversion*

- Expanding each octal digit into three bits

□ *Hexadecimal to binary conversion*

- Expanding each hexadecimal digit into four bits

Conversion between any two bases, other than decimal (Special cases)

- Example 13: Convert 743_8 to binary

743_8

→ $111\ 100\ 011_2$

→ 111100011_2

0 = 000

1 = 001

2 = 010

3 = 011

4 = 100

5 = 101

6 = 110

7 = 111

Conversion between any two bases, other than decimal (Special cases)

- Example 14: Convert $FA9_{16}$ to binary

$FA9_{16}$

→ $1111\ 1010\ 1001_2$

→ 111110101001_2

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

Conversion between any two bases, other than decimal (Special cases)

■ Octal to hexadecimal or hexadecimal to octal:

- Convert from the **source** base to the **binary**
 - Expanding each digit into three bits (in case of octal) or four bits (in case of hexadecimal)
- Convert from the **binary** to the **destination** base
 - Group bits in three's (in case of octal) or four's (in case of hexadecimal), starting from the binary point (*pad the last group from both sides with 0's, if needed*)

Conversion between any two bases, other than decimal (Special cases)

- Example 15: Convert ABC_{16} to octal

ABC_{16}

→ $1010\ 1011\ 1100_2$

→ 101010111100_2

→ $101\ 010\ 111\ 100_2$

→ 5274_8

0 = 000

1 = 001

2 = 010

3 = 011

4 = 100

5 = 101

6 = 110

7 = 111

0 = 0000

1 = 0001

2 = 0010

3 = 0011

4 = 0100

5 = 0101

6 = 0110

7 = 0111

8 = 1000

9 = 1001

A = 1010

B = 1011

C = 1100

D = 1101

E = 1110

F = 1111

Conversion between any two bases, other than decimal (Special cases)

- Example 16: Convert $0.AB_{16}$ to octal

$0.AB_{16}$

→ $0.1010\ 1011_2$

→ 0.10101011_2

→ $00.101\ 010\ 110_2$

→ 0.526_8

0 = 000

1 = 001

2 = 010

3 = 011

4 = 100

5 = 101

6 = 110

7 = 111

0 = 0000

1 = 0001

2 = 0010

3 = 0011

4 = 0100

5 = 0101

6 = 0110

7 = 0111

8 = 1000

9 = 1001

A = 1010

B = 1011

C = 1100

D = 1101

E = 1110

F = 1111

Conversion between any two bases, other than decimal (Special cases)

- Example 17: Convert $AB.BA_{16}$ to octal

$AB.BA_{16}$

→ $1010\ 1011.1011\ 1010_2$

→ 10101011.1011101_2

→ $010\ 101\ 011.101\ 110\ 100_2$

→ 253.564_8

0 = 000

1 = 001

2 = 010

3 = 011

4 = 100

5 = 101

6 = 110

7 = 111

0 = 0000

1 = 0001

2 = 0010

3 = 0011

4 = 0100

5 = 0101

6 = 0110

7 = 0111

8 = 1000

9 = 1001

A = 1010

B = 1011

C = 1100

D = 1101

E = 1110

F = 1111

Conversion between any two bases, other than decimal (Special cases)

- Example 18: Convert 123_8 to hexadecimal

123_8

→ $001\ 010\ 011_2$

→ $\quad\quad 1010011_2$

→ $0101\ 0\ 011_2$

→ 53_{16}

0 = 000

1 = 001

2 = 010

3 = 011

4 = 100

5 = 101

6 = 110

7 = 111

0 = 0000

1 = 0001

2 = 0010

3 = 0011

4 = 0100

5 = 0101

6 = 0110

7 = 0111

8 = 1000

9 = 1001

A = 1010

B = 1011

C = 1100

D = 1101

E = 1110

F = 1111

Conversion between any two bases, other than decimal (Special cases)

- Example 19: Convert 0.123_8 to hexadecimal

0.123_8

→ $0.001\ 010\ 011_2$

→ 0.001010011_2

→ $0000.0010\ 1001\ 1000_2$

→ 0.298_{16}

0 = 000

1 = 001

2 = 010

3 = 011

4 = 100

5 = 101

6 = 110

7 = 111

0 = 0000

1 = 0001

2 = 0010

3 = 0011

4 = 0100

5 = 0101

6 = 0110

7 = 0111

8 = 1000

9 = 1001

A = 1010

B = 1011

C = 1100

D = 1101

E = 1110

F = 1111

Conversion between any two bases, other than decimal (Special cases)

- Example 20: Convert 321.123_8 to hexadecimal

321.123_8

→ $011\ 010\ 001.001\ 010\ 011_2$

→ 11010001.001010011_2

→ $1101\ 0001.0010\ 1001\ 1000_2$

→ $D1.298_{16}$

0 = 000

1 = 001

2 = 010

3 = 011

4 = 100

5 = 101

6 = 110

7 = 111

0 = 0000

1 = 0001

2 = 0010

3 = 0011

4 = 0100

5 = 0101

6 = 0110

7 = 0111

8 = 1000

9 = 1001

A = 1010

B = 1011

C = 1100

D = 1101

E = 1110

F = 1111