CSC 3210 Computer organization and programming

Chapter 3 & Quiz3

Chunlan Gao

Georgia State University

Attendance Number

- The attendance number is

9978

Review

7/2=3....1

Iteration	Step	Quotient	Divisor	Remainder
0	Initial values	0000	0010 0000	0000 0111
1	1: Rem = Rem – Div	0000	0010 0000	1110 0111
	2b: Rem < 0 ⟹ +Div, SLL Q, Q0 = 0	0000	0010 0000	0000 0111
	3: Shift Div right	0000	0001 0000	0000 0111
2	1: Rem = Rem – Div	0000	0001 0000	1111 0111
	2b: Rem < 0 ⟹ +Div, SLL Q, Q0 = 0	0000	0001 0000	0000 0111
	3: Shift Div right	0000	0000 1000	0000 0111
3	1: Rem = Rem – Div	0000	0000 1000	1111 1111
	2b: Rem < 0 ⟹ +Div, SLL Q, Q0 = 0	0000	0000 1000	0000 0111
	3: Shift Div right	0000	0000 0100	0000 0111
4	1: Rem = Rem – Div	0000	0000 0100	0000 0011
	2a: Rem ≥ 0 ⟹ SLL Q, Q0 = 1	0001	0000 0100	0000 0011
	3: Shift Div right	0001	0000 0010	0000 0011
5	1: Rem = Rem – Div	0001	0000 0010	0000 0001
	2a: Rem ≥ 0 ⟹ SLL Q, Q0 = 1	0011	0000 0010	0000 0001
	3: Shift Div right	0011	0000 0001	0000 0001

FIGURE 3.10 Division example using the algorithm in Figure 3.9. The bit examined to determine the next step is circled in color.

RISC-V Division

- Four instructions:
  - div, rem: signed divide, remainder
  - divu, remu: unsigned divide, remainder

Floating Point

- Representation for non-integral numbers
  - Including very small and very large numbers
- Like scientific notation
  - $-2.34\,0753 \times 10^{56}$
  - $+0.00256779 \times 10^{-4}$
  - $+987.0245664 \times 10^{9}$
- In binary
  - $\pm 1.xxxxxxx_2 \times 2^{yyyy}$
- Types **`float` and `double`** in C

normalized

unnormalized

$+1.10101 \times 2^{-2} \rightarrow +0.0110101$

$-1.10101 \times 2^{2} \rightarrow -110.101$

Floating-Point Representation

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s	exponent								fraction																						
1 bit	8 bits								23 bits																						

overflow (floating-point) A situation in which a positive exponent becomes too large to fit in the exponent field.

underflow (floating-point) A situation in which a negative exponent becomes too large to fit in the exponent field.

Floating-point numbers are of the form

$(-1)^{S} \times F \times 2^E$

F： involves the value in the fraction field E： involves the value in the exponent field

Floating Point Standard

- Defined by IEEE Std 754-1985
- Developed in response to divergence of representations
  - Portability issues for scientific code
- Now almost universally adopted
- Two representations
  - Single precision (32-bit)
  - Double precision (64-bit)

IEEE Floating-Point Format

Type(precision)	Sign	Exponent	Fraction
Single	1bit	8 bits	23 bits
Double	1bit	23 bits	52 bits

$$(-1)^S \times (1 + \text{Fraction}) \times 2^E \qquad (-1)^S \times (1 + (s1 \times 2^{-1}) + (s2 \times 2^{-2}) + (s3 \times 2^{-3}) + (s4 \times 2^{-4}) + \ldots) \times 2^E$$

- S: sign bit (0 $\Rightarrow$ non-negative, 1 $\Rightarrow$ negative)
- Normalize significand: 1.0 ≤ |significand| < 2.0
  - Always has a leading pre-binary-point 1 bit, so no need to represent it explicitly (hidden bit)
  - Significand is Fraction with the “1.” restored
- Exponent: excess representation: actual exponent + Bias
  - Ensures exponent is unsigned
  - Single: Bias = 127; Double: Bias = 1203

Single-Precision Range

- Exponents 00000000 and 11111111 reserved
- Smallest value
  - Exponent: 00000001
    $\Rightarrow$ actual exponent = 1 – 127 = –126
  - Fraction: 000…00 $\Rightarrow$ significand = 1.0
  - $\pm 1.0 \times 2^{-126} \approx \pm 1.2 \times 10^{-38}$(ten)
- Largest value
  - exponent: 11111110
    $\Rightarrow$ actual exponent = 254 – 127 = +127
  - Fraction: 111…11 $\Rightarrow$ significand ≈ 2.0
  - $\pm 2.0 \times 2^{+127} \approx \pm 3.4 \times 10^{+38}$

geometric sequence：

$$S_n = \frac{a_1(1-q^n)}{1-q}(q \neq 1)$$

$$Sn = \frac{1/2\ (1-\frac{1}{2} \wedge n)}{1-1/2} = 1$$

Double-Precision Range

- Exponents 0000…00 and 1111…11 reserved
- Smallest value
  - Exponent: 00000000001
    $\Rightarrow$ actual exponent = 1 – 1023 = –1022
  - Fraction: 000…00 $\Rightarrow$ significand = 1.0
  - $\pm 1.0 \times 2^{-1022} \approx \pm 2.2 \times 10^{-308}$
- Largest value
  - Exponent: 11111111110
    $\Rightarrow$ actual exponent = 2046 – 1023 = +1023
  - Fraction: 111…11 $\Rightarrow$ significand ≈ 2.0
  - $\pm 2.0 \times 2^{+1023} \approx \pm 1.8 \times 10^{+308}$

Floating-Point Example

- Represent –0.75 $0.75=3/4=11/(2^2)=0.11*2^{-2}=1.1*2^{-1}$
  - $-0.75 = (-1)^1 \times 1.1_2 \times 2^{-1}$
  - S = 1
  - Fraction = $1000...00_2$
  - Exponent = –1 + Bias
    - Single: –1 + 127 = 126 = 01111110_2
    - Double: –1 + 1023 = 1022 = 01111111110_2
- Single: 1011111101000...00
- Double: 1011111111101000...00

Floating-Point Example 2

- Represent 0.1ten ? (single- precision)
- 0.00011001100110011 two
- 3 d c c c c c d (IEEE format)

Floating-Point Example

- What number is represented by the **single-precision** float

  1 10000001 01000...00

  - S = 1
  - Fraction = $01000...00_2$
  - Exponent = $10000001_2 = 129$
- $x = (-1)^1 \times (1 + 0*2^{-1}+1*2^{-2}) \times 2^{(129-127)}$

  $= (-1) \times 1.25 \times 2^2$

  $= -5.0$

Floating-Point Example

- 0 10000001 10100000000000000000000 (**single-precision)**

- **Sign (1 bit) = 0** (positive number)
- **Exponent (8 bits) = 10000001** (decimal 129),

  actual exponent = 129 - 127 = 2
- **Fraction (23 bits) = 10100000000000000000000**
- **Significand = 1.101** (the **implicit 1 is added to the fraction**)
- **Final computed value:** $1.101 \times 2^2 = 110.1_2 = 6.5_{10}$