

Floating point

Lecture B04



Lecture notes section B04

2002-01-11

CSE1303 Part B lecture notes

1

Last time

- Arithmetic
 - unsigned addition
- Signed integers
 - two's complement system
- More arithmetic
 - signed addition
 - subtraction
 - multiplication

2002-01-11

CSE1303 Part B lecture notes

2

In this lecture

- Fixed point
- Scientific notation
- Floating point
- Anatomy of floating point
 - sign
 - exponent
 - mantissa
- Floating point arithmetic
 - multiplication
 - addition
- Limitations of floating point

2002-01-11

CSE1303 Part B lecture notes

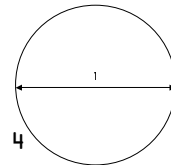
3

Geometry test

Calculate the area of this circle:

$$\begin{aligned} \text{area} &= \pi d^2 / 4 \\ &= 3.142 \times 1 \times 1 / 4 \\ &= 0.785 \end{aligned}$$

$$\begin{aligned} \text{area} &= \pi * d * d / 4 \\ &= 3 * 1 * 1 / 4 \\ &= 0 \end{aligned}$$



2002-01-11

CSE1303 Part B lecture notes

4

Real number representation

- Real numbers
 - numbers which are not necessarily integers
 - have an integer part and fractional part
- Need a way to represent (or approximate) real numbers using binary

2002-01-11

CSE1303 Part B lecture notes

5

Rational numbers

- Attempt 1: rational numbers
 - already know how to represent integers
 - express numbers as ratio of two integers
 - numerator and denominator
 - $9.75 = 39 \div 4$
 - $0 = 0 \div 1$
 - $-42 = -42 \div 1$
 - $\pi \approx 22 \div 7$
 - to represent a rational number, just need to specify both integers

2002-01-11

CSE1303 Part B lecture notes

6

Rational numbers

- Problems with rational number representations
 - multiple representations of the same value
 - $9.75 = 39 \div 4 = 975 \div 1000 = -117 \div -12 = \dots$
 - $-42 = -42 \div 1 = 42 \div -1 = \dots$
 - $0 = 0 \div 1 = 0 \div 2 = 0 \div -1 = \dots$
 - makes comparing values difficult
 - no exact representations of some values
 - $\pi \approx 22 \div 7 \approx 335 \div 113 \approx 314159 \div 100000 = \dots$
 - not possible for irrational numbers anyway
 - decimal has same problem

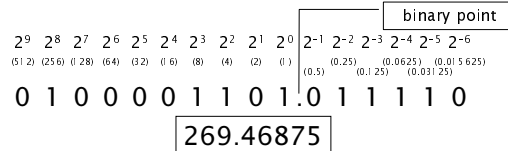
2002-01-11

CSE1303 Part B lecture notes

7

Fixed point

- Attempt 2: fixed point
 - insert implicit "binary point" between two bits
 - bits to left of point have value ≥ 1
 - bits to right of point have value < 1



2002-01-11

CSE1303 Part B lecture notes

8

Fixed point

- Problems with fixed point representations
 - small range of numbers
 - smallest number 0.015625
 - largest number 1023.984375
 - fixed point means wasted bits
 - π represented as 3.140625
 - 8 most significant bits all 0

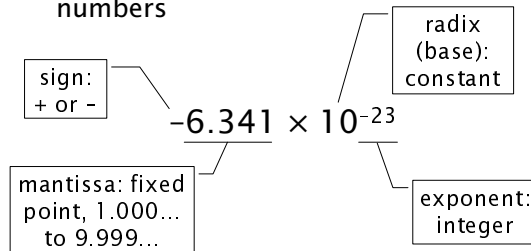
2002-01-11

CSE1303 Part B lecture notes

9

Scientific notation

- Used to represent a wide range of numbers



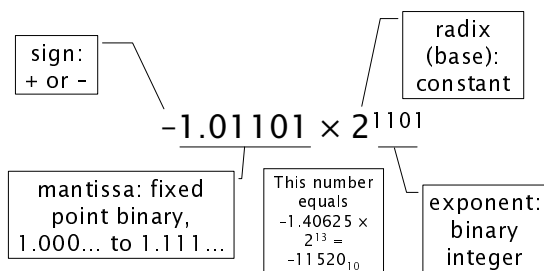
2002-01-11

CSE1303 Part B lecture notes

10

Scientific notation

- Same idea works in binary



2002-01-11

CSE1303 Part B lecture notes

11

Scientific notation

- Advantages
 - very wide range of representable numbers
 - limited by range of exponent
 - similar precision for all values
 - no wasted bits
- Disadvantages
 - some values still not exactly representable
 - e.g., π

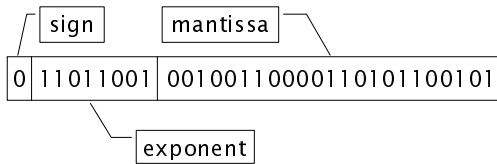
2002-01-11

CSE1303 Part B lecture notes

12

Floating point

- Binary representation using scientific notation
 - IEEE754 standard



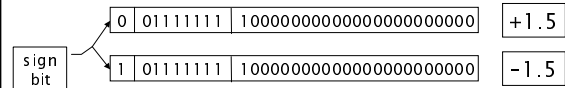
2002-01-11

CSE1303 Part B lecture notes

13

Floating point

- Sign
 - one bit
 - signed magnitude representation
 - 0 \Leftrightarrow number is positive
 - 1 \Leftrightarrow number is negative



2002-01-11

CSE1303 Part B lecture notes

14

Floating point

- Exponent
 - could be represented using two's complement signed notation
 - instead represented using excess-k notation
 - represented value in exponent field is k more than intended value
 - k is constant (for C float, $k = 127$)
 - exponent 0000001 (1_{10}) is -126
 - exponent 01111111 (127_{10}) is 0
 - exponent 11111110 (254_{10}) is $+127$
 - exponent 000...000 and 111...111 reserved for special meanings
 - all bits 0: denormalized numbers and zero
 - all bits 1: infinity and not-a-number (indeterminate)

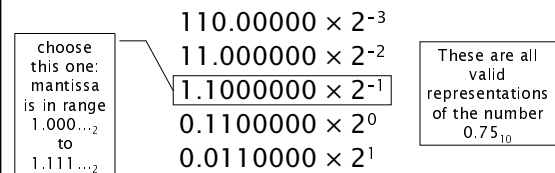
2002-01-11

CSE1303 Part B lecture notes

15

Floating point

- Number is normalized
 - exponent is chosen such that $1_{10} \leq \text{mantissa} < 2_{10}$



2002-01-11

CSE1303 Part B lecture notes

16

Floating point

- Mantissa
 - represented as fixed-point value between $1.000..._2$ and $1.111..._2$
 - first bit (before the point) is always 1
 - don't waste a bit to store it in number
 - fixed precision
 - for C float, 23 bits available
 - plus 1 implicit bit
 - 24 significant bits
 - mantissa sometimes called significand

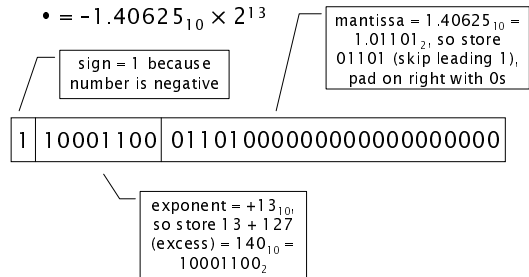
2002-01-11

CSE1303 Part B lecture notes

17

Floating point

- Example: -11520_{10} as a C float
 - $= -1.40625_{10} \times 2^{13}$



2002-01-11

CSE1303 Part B lecture notes

18

Floating point

- C has floating point types of various sizes
- 32 bits (float)
 - 1 bit sign, 8 bits exponent, 23 (24) bits mantissa, excess 127
- 64 bits (double)
 - 1 bit sign, 11 bits exponent, 52 (53) bits mantissa, excess 1023
- 80 bits (long double)
 - 1 bit sign, 15 bits exponent, 64 bits mantissa, excess 16383

2002-01-11

CSE1 303 Part B lecture notes

19

Floating point

- C does many calculations internally using doubles
 - most modern computers can operate on doubles as fast as on floats
 - may be less efficient to use float variables, even though smaller in size
 - long double operations may be very slow
 - may have to be implemented by software
- Using literal floating point values in C
 - require decimal point
 - 5 is of type int, use 5.0 if you want floating point
 - optional exponent
 - 5.0e-12 means 5.0×10^{-12} (decimal)

2002-01-11

CSE1 303 Part B lecture notes

20

Limitations of floating point

- Size of exponent is fixed
 - cannot represent very large numbers
 - for C float, exponent is 8 bits, excess is 127
 - largest exponent is +127 (254_{10} (11111110_2) - 127)
 - 11111111 reserved for infinity and not-a-number (NaN)
 - largest representable numbers
 - positive: $1.1111..._2 \times 2^{127} = 3.403..._{10} \times 10^{38}$
 - negative: $-1.1111..._2 \times 2^{127} = -3.403..._{10} \times 10^{38}$
 - overflow occurs if numbers larger than this are produced
 - rounds up to \pm infinity
 - solution: use a floating point format with a larger exponent
 - double (11 bits), long double (15 bits)

2002-01-11

CSE1 303 Part B lecture notes

21

Limitations of floating point

- Size of exponent is fixed
 - cannot represent very small numbers
 - for C float, exponent is 8 bits, excess is 127
 - smallest exponent is -126 (1_{10} (0000001_2) - 127)
 - 00000000 reserved for zero and denormalized numbers
 - smallest representable (normalized) numbers
 - positive: $1.000..._2 \times 2^{-126} = 1.175..._{10} \times 10^{-38}$
 - negative: $-1.000..._2 \times 2^{-126} = -1.175..._{10} \times 10^{-38}$
 - underflow occurs if numbers smaller than this are produced
 - rounds down to zero (also denormalized)
 - solution: use a floating point format with a larger exponent
 - double (11 bits), long double (15 bits)

2002-01-11

CSE1 303 Part B lecture notes

22

Limitations of floating point

- Size of mantissa is fixed
 - limited precision in representations
 - C float has 23 (24) bits of mantissa
 - smallest possible change in a number is to toggle LSB
 - place value $2^{-23} \approx 1.2 \times 10^{-7}$
 - C float has (almost) 7 decimal digits of precision
 - solution: use a floating point format with a larger mantissa
 - double (53 bits), long double (64 bits)

2002-01-11

CSE1 303 Part B lecture notes

23

Limitations of floating point

- Size of mantissa is fixed
 - some values cannot be represented exactly
 - e.g. $1/3_{10} = 0.0101010101010101..._2$
 - continuing fraction never ends
 - cannot fit in 24 (or 240 (or 24000)) bits
 - solution: none, same problem occurs in decimal scientific notation
 - can use higher precision floating point type to improve accuracy
 - if exact representation is needed, use rational numbers

2002-01-11

CSE1 303 Part B lecture notes

24

Floating point arithmetic

- Multiplication
 - this example in decimal
 - same method in binary

$+8.19 \times 10^9 \times -2.35 \times 10^{12}$

step 1: sign

$\begin{matrix} + \times + = + \\ - \times + = - \\ + \times - = - \\ - \times - = + \end{matrix}$

$-$

2002-01-11 CSEI 303 Part B lecture notes 25

Floating point arithmetic

- Multiplication
 - this example in decimal
 - same method in binary

$+8.19 \times 10^9 \times -2.35 \times 10^{12}$

step 2: exponent

add exponents

$- \times 10^{21}$

2002-01-11 CSEI 303 Part B lecture notes 26

Floating point arithmetic

- Multiplication
 - this example in decimal
 - same method in binary

$+8.19 \times 10^9 \times -2.35 \times 10^{12}$

step 3: mantissa

multiply mantissas

-19.2465×10^{21}

2002-01-11 CSEI 303 Part B lecture notes 27

Floating point arithmetic

- Multiplication
 - this example in decimal
 - same method in binary

-19.2465×10^{21}

step 4: renormalize and round

-1.92×10^{22}

some loss of precision is inevitable in floating-point multiplication

2002-01-11 CSEI 303 Part B lecture notes 28

Floating point arithmetic

- Addition
 - this example in decimal
 - same method in binary

$+9.35 \times 10^5 + +8.14 \times 10^4$

step 1: sign and operation

signs same: addition

use signs and magnitudes of numbers to determine sign and whether operation is effectively addition or subtraction

$+$

2002-01-11 CSEI 303 Part B lecture notes 29

Floating point arithmetic

- Addition
 - this example in decimal
 - same method in binary

$+9.35 \times 10^5 + +8.14 \times 10^4$

step 2: match exponents

rewrite smaller number so that it has same exponent as larger number

$+0.814 \times 10^5$

2002-01-11 CSEI 303 Part B lecture notes 30

Floating point arithmetic

- Addition
 - this example in decimal
 - same method in binary

$$+9.35 \times 10^5 + +0.814 \times 10^5$$

step 3: exponent

copy exponent

$$+ \quad \times 10^5$$

2002-01-11 CSEI 303 Part B lecture notes 31

Floating point arithmetic

- Addition
 - this example in decimal
 - same method in binary

$$+9.35 \times 10^5 + +0.814 \times 10^5$$

step 4: mantissa

add mantissas

$$+10.164 \times 10^5$$

2002-01-11 CSEI 303 Part B lecture notes 32

Floating point arithmetic

- Addition
 - this example in decimal
 - same method in binary

$$+10.164 \times 10^5$$

step 5: normalize and round

$$+1.02 \times 10^6$$

loss of precision is likely if numbers are of very different magnitudes

2002-01-11 CSEI 303 Part B lecture notes 33

Limitations of floating point

- Addition of floating point numbers not associative
 - $A + (B + C) \neq (A + B) + C$
 - if numbers significantly different magnitudes
 - for instance:
 - $A = 1.00 \times 10^3$
 - $B = 4.00 \times 10^0$
 - $C = 3.00 \times 10^0$
 - 3 significant digits

2002-01-11 CSEI 303 Part B lecture notes 34

Limitations of floating point

- $A + (B + C)$

$$\begin{array}{r} B: 4.00 \times 10^0 \\ + C: 3.00 \times 10^0 \\ \hline 7.00 \times 10^0 \\ \downarrow \\ 0.007 \times 10^3 \\ + A: 1.00 \times 10^3 \\ \hline 1.007 \times 10^3 \end{array}$$

result of addition B + C

rewrite sum to add to A

result after rounding: 1.01×10^3

2002-01-11 CSEI 303 Part B lecture notes 35

Limitations of floating point

- $(A + B) + C$

$$\begin{array}{r} B: 4.00 \times 10^0 \\ \downarrow \\ 0.004 \times 10^3 \\ + A: 1.00 \times 10^3 \\ \hline 1.004 \times 10^3 \\ \downarrow \\ 1.00 \times 10^3 \end{array}$$

rewrite B to add to A

result of sum A + B

result after rounding, carry forward to next addition

2002-01-11 CSEI 303 Part B lecture notes 36

Limitations of floating point

- (A + B) + C (continued)

$$\begin{array}{r} C: 3.00 \times 10^0 \\ \downarrow \\ 0.003 \times 10^3 \\ 1.00 \times 10^3 \\ \hline 1.003 \times 10^3 \\ \downarrow \\ 1.00 \times 10^3 \end{array}$$

rewrite C to add to sum

carried forward from sum on previous slide

result after rounding:
 1.00×10^3

2002-01-11

CSE1 303 Part B lecture notes

37

Covered in this lecture

- Floating point
 - sign
 - exponent
 - mantissa
- Floating point arithmetic
 - multiplication
 - addition
- Limitations of floating point
 - limited precision
 - overflow and underflow
 - addition not associative

2002-01-11

CSE1 303 Part B lecture notes

38

Going further

- Floating point subtraction and division
- Infinities, Not-a-Number and denormalized numbers
 - IEEE754's dustier corners

2002-01-11

CSE1 303 Part B lecture notes

39

Next time

- Bit manipulation
 - bitwise operations
 - shifting
 - masking



Reading:
Lecture notes section B05

2002-01-11

CSE1 303 Part B lecture notes

40

Copyright

Copyright © 2001 Deborah Pickett.
No part of this presentation may be duplicated without permission from the author.

2002-01-11

CSE1 303 Part B lecture notes

41