CS2100 COMPUTER ORGANISATION

http://www.comp.nus.edu.sg/~cs2100/

Lecture #3d

Data Representation and Number Systems

National University of Singapore

School of Computing

Questions?

Ask at https://app.sli.do/event/qVCWNryB45Bnh6p2HRfnFG

OR

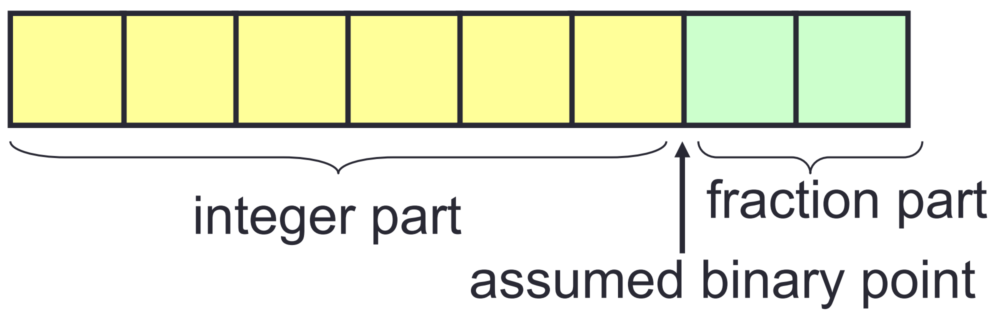
Scan and ask your questions here!
(May be obscured in some slides)

11. Real Numbers

- Many applications involve computations not only on integers but also on real numbers.
- How are real numbers represented in a computer system?
- Due to the finite number of bits, real number are often represented in their approximate values.

11.1 Fixed-Point Representation

- In fixed-point representation, the number of bits allocated for the whole number part and fractional part are fixed.
- For example, given an 8-bit representation, 6 bits are for whole number part and 2 bits for fractional parts.



- If 2s complement is used, we can represent values like:

$$011010.11_{2s} = 26.75_{10}$$

$$111110.11_{2s} = -000001.01_{2} = -1.25_{10}$$

11.2 Floating-Point Representation (1/4)

- Fixed-point representation has limited range.
- Alternative: Floating point numbers allow us to represent very large or very small numbers.
- Examples:

  0.23×10^{23} (very large positive number)

  0.5×10^{-37} (very small positive number)

  -0.2397×10^{-18} (very small negative number)

11.2 IEEE 754 Floating-Point Rep. (2/4)

- 3 components: **sign**, **exponent** and **mantissa (fraction)**

sign	exponent	mantissa

- The base (radix) is assumed to be 2.
- Two formats:
  - Single-precision (32 bits): 1-bit sign, 8-bit exponent with bias 127 (excess-127), 23-bit mantissa
  - Double-precision (64 bits): 1-bit sign, 11-bit exponent with bias 1023 (excess-1023), and 52-bit mantissa
- We will focus on the single-precision format
- Reading
  - DLD pages 32 - 33
  - IEEE standard 754 floating point numbers: http://steve.hollasch.net/cgindex/coding/ieeefloat.html

11.2 IEEE 754 Floating-Point Rep. (3/4)

- 3 components: **sign**, **exponent** and **mantissa (fraction)**

sign	exponent	mantissa

- Sign bit: 0 for positive, 1 for negative.
- Mantissa is **normalised** with an implicit leading bit 1
  - 110.1_2 → normalised → $1.101_2 \times 2^2$ → only **101** is stored in the mantissa field
  - 0.00101101_2 → normalised → $1.01101_2 \times 2^{-3}$ → only **01101** is stored in the mantissa field

11.2 IEEE 754 Floating-Point Rep. (4/4)

- Example: How is -6.5_{10} represented in IEEE 754 single-precision floating-point format?

$$-6.5_{10} = -110.1_2 = -1.\boxed{101}_2 \times 2^{2}$$

Exponent = 2 + 127 = 129 = 10000001_2

sign	exponent (excess-127)	mantissa
1	10000001	10100000000000000000000

- We may write the 32-bit representation in hexadecimal:

$1\ 10000001\ 10100000000000000000000_2 = \text{C0D00000}_{16}$

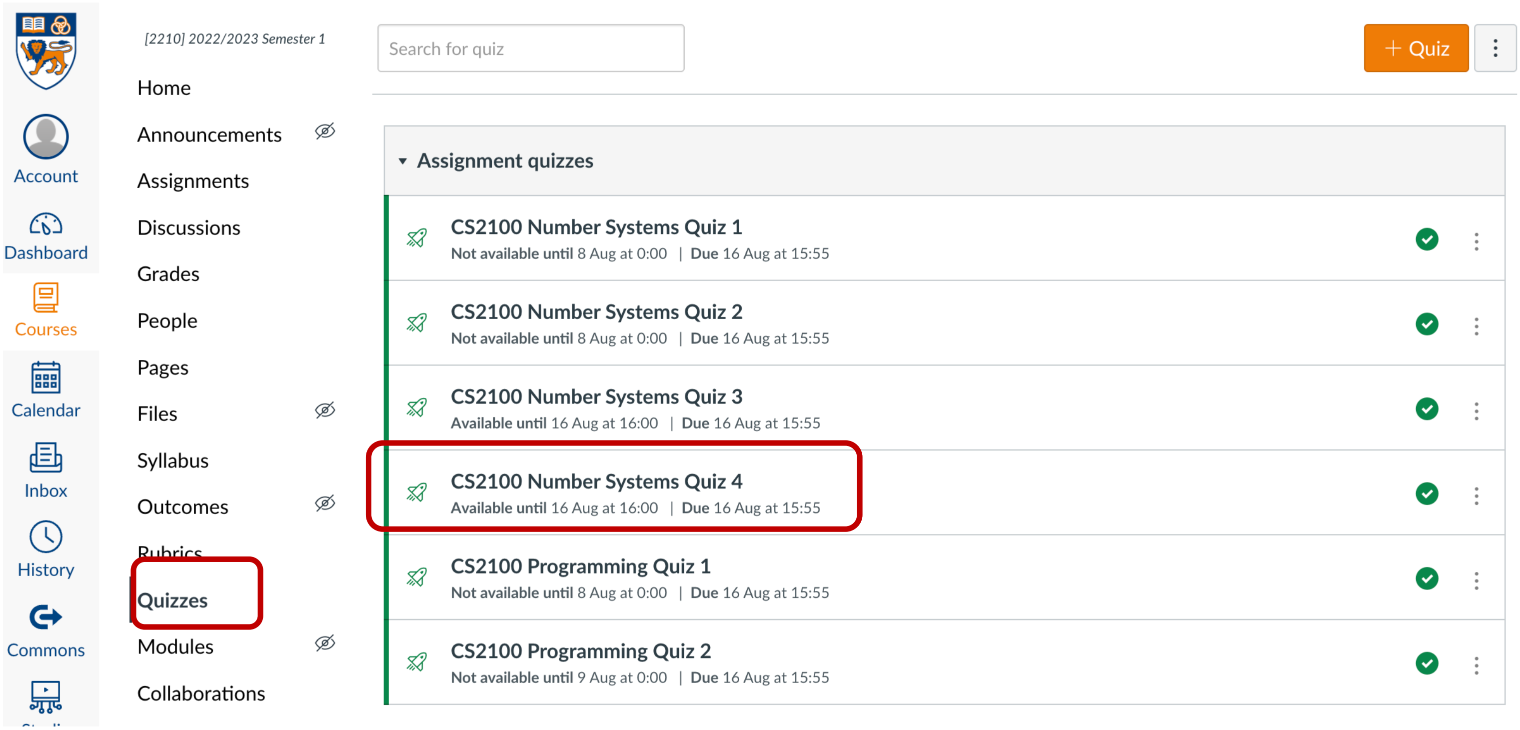
(Slide 4)

As an 'int', it is -1060110336

As an 'float', it is -6.5

Quiz

- Please complete the “CS2100 C Number Systems Quiz 4” in Canvas.
  - Access via the “Quizzes” tool in the left toolbar and select the quiz on the right side of the screen.



End of File