Final Exam

- In-person exam

  • A longer quiz covers multiple topics. Five questions, and each has several sub questions.
  
  • The final exam will be at **9:00 am Sunday, Dec. 17 @ DIAM 142**.
  
  • The exam should take you about 1.5 hours, but you have 3 hours for it.
  
  • Close notes and close books. You can bring **one letter size cheatsheet** (both sides, handwritten or typed) and **a calculator**. No other electronic device.
  
  • You are expected to take the final exam individually and **sit apart from your neighbors**, at least one empty seat between you and each of your left and right neighbors.
  
  • It’s your responsibility to make your answers readable. All answers should be brief but clear. Make sure your answers are neat if you wish full credit. I will not take off points for trivial errors, such as misspelling. However, I will take off points if your answers are messy and difficult to read.

Wrapping up …

- Binary/Decimal

  • conversion between them
  
  • 2’ complement

  \[
  \begin{array}{ccccccc}
    -2^7 & 2^6 & 2^5 & 2^4 & 2^3 & 2^2 & 2^1 & 2^0 \\
  \end{array}
  \]

- Data representations

  • Octal: 3 bits per digit
  
  • Hex: 4 bits per digit (conversion between hex and binary)
    
    - The last question of Q2
- Digital logic
  - gates (AND, OR, NOT, NAND, NOR, XOR), their uses and implementations
  - circuit design (k-map)
    - truth table from question specification, k-map (00, 01, 11, 10), sum of produce
    - HW1 and Q1 (2’s complement and circuit design)
  - useful combinational circuits: MUX, DEMUX, and Decoder.
  - VHDL
  - sequential circuits: flipflops, registers
  - difference between combinational circuits and sequential circuits
    - Lecture notes: sequential circuits
  - state machines
  - programmable circuits

- Memory
  - memory hierarchy (registers, cache, main memory)
  - mapping functions: direct, associative, set associative; addressable unit: word, byte
  - average access-time, hit-rate
    - hit rate: the ratio that the accessed word is found in the faster memory
  - HW5, Q5

- ISA
  - stack architectures
    - both operands of ALU popped from the top of the stack, result of ALU push into the top of the stack

  **PUSH A** # push value at location A to the top of the stack
  **PUSH B** # push value at location B to the top of the stack
  **ADD** # add the top two elements of the stack and save the result back to the top of the stack
POP C # store the value at the top of the stack (which is the result) to location C

- **addressing modes** (immediate, direct, indirect, register direct, register indirect, displacement, advantages/disadvantages)

- Lecture notes for stack architectures and addressing modes

- **Assembly language**
  - the benefits of assembly language
  - two-pass assembler
  - assembly program (Examples on the notes)

- **Pipeline and superscalar**
  - pipeline improves the throughput of a processor
  - superscalar supports multiple pipelines in a processor
  - **dependencies** (RAW, WAW, WAR), possible way to address WAW and WAR dependencies (register renaming)
  - instruction issue policies (out-of-order issue, out-of-order completion)

- HW8, Q8