Questions for the exam in "Computers and Networks"

Chapter 1 Introduction

1.1 Name the three basic components of every computer.
1.2 What units are typically used to measure the speed of a computer clock?
1.3 State Moore's Law.
1.4 What was it about the Von Neumann architecture that distinguished it from its predecessors?
1.5 How does the fetch-decode-execute cycle work?
1.6 What is the underlying premise of Amdahl's law?

Chapter 2 Data Representation in Computer Systems

2.1 Explain how the terms bit, byte, nibble, and word are related.
2.2 Name the three ways in which signed integers can be represented in digital computers and explain the differences. Which one of the three representations for signed integers is used most often by digital computer systems?
2.3 What is overflow and how can it be detected?
2.4 What is the goal of Booth's algorithm?
2.5 What are the three components of floating-point representation?
2.6 What is normalization in the context of floating-point representation and why is it necessary?
2.7 How many bits does Unicode require? Why was Unicode created?
2.8 What is a Hamming code?

Chapter 3 Boolean Algebra and Digital Logic

3.1 Which Boolean operation is referred to as a Boolean product? Which Boolean operation is referred to as a Boolean sum?
3.2 Why is it important for Boolean expressions to be minimized in the design of digital circuits?
3.3 Name the four basic logic gates.
3.4 What are the two universal gates named in the textbook? Why are these universal gates important?
3.5 Describe the operation of a ripple-carry adder. Why are ripple-carry adders not used in most computers today?
3.6 What do we call a circuit that takes several inputs and their respective values to select one specific output line? Name one important application of these devices.
3.7 What kind of circuit selects binary information from one of many input lines and direct it to a single output line? Name an application of these devices.
3.8 How are sequential circuits different from combinational circuits?
3.9 What is the basic element of a sequential circuit?
3.10 How is a JK flip-flop related to a SR flip-flop? Why are JK flip-flops often preferred to SR flip-flops?
3.11 Which flip-flop gives a true representation of computer memory?

Chapter 4 MARIE: An Introduction to a Simple Computer

4.1 What is the function of a CPU?
4.2 What does the control unit do?
4.3 Where are the registers located and what are the different types?
4.4 What is the function of the ALU? How does the ALU know which function to perform?
4.5 Why is a bus often a communication bottleneck?
4.6 Explain the difference between address buses, data buses, and control buses.
4.7 How do CPU clocks and bus clocks differ?
4.8 Explain the difference between memory-mapped I/O and instruction-based I/O.
4.9 Describe how interrupts work. Name some reasons for triggering an interrupt.
4.10 What is an opcode?
4.11 How does machine language differ from assembly language?
4.12 Is a micro-operation the same thing as a machine instruction?
4.13 How does interrupt-driven I/O work?
4.14 Explain how an assembler works, including how it generates the symbol table, what it does with source and object code, and how it handles labels.
4.15 Explain the difference between hardwired and micro-programmed control.
4.16 Compare RISC machines to CISC machines.

Chapter 5 A Closer Look at Instruction Set Architectures

5.1 Explain the difference between memory-memory, register-memory, and load-store computer architectures.
5.2 Several design decisions exist with regards to instruction sets. Name some and explain.
5.3 If a byte-addressable machine with 32-bit words stores a hex value of 98765432, indicate how this value would be stored on a big endian machine and a little endian machine.
5.4 Why might stack architectures represent arithmetic expressions in reverse Polish notation?
5.5 Name the seven types of data instructions and explain each.
5.6 What is an addressing mode? Give examples of immediate, direct, register, indirect, register indirect, indexed addressing, and based addressing.
5.7 Explain the concept behind instruction pipelining. What are the pipeline conflicts that can cause a slowdown in the pipeline?

Chapter 6 Memory

6.1 What is SRAM and DRAM? Which is faster?
6.2 What are advantages of using DRAM for memory instead of SRAM?
6.3 What is ROM? Name some applications where ROMs are often used?
6.4 Explain the concept of locality of reference and state its importance to memory systems.
6.5 What are the three forms of locality?
6.6 Which of L1 or L2 cache is faster? Which is smaller?
6.7 What are the three fields in a direct mapped cache address? How are they used to access a word located in the cache?
6.8 How does associative memory differ from regular memory?
6.9 Explain how fully associative cache is different from direct mapped cache.
6.10 Explain some of the cache replacement policies presented in the textbook.
6.11 When does caching behave badly?
6.12 What is the difference between a unified cache and a Harvard cache?

6.13 What is the difference between a virtual memory address and a physical memory address? Which is larger? Why?
6.14 What is the purpose of virtual memory? What are the advantages and disadvantages of virtual memory implemented through paging?
6.15 What is paging?
6.16 What is a page fault?
6.17 What are the components (fields) of a virtual address?

Chapter 7 Input/Output and Storage Systems
7.1 State Amdahl's Law in words.
7.2 Explain how programmed I/O is different from interrupt-driven I/O.
7.3 How is channel I/O different from interrupt-driven I/O?
7.4 What is polling?
7.5 What distinguishes an asynchronous bus from a synchronous bus?

Chapter 8 System Software
8.1 Describe how multiprogramming systems differ from timesharing systems.
8.2 Describe the two divergent philosophies concerning operating system kernel design.
8.3 What are the benefits and drawbacks to a GUI operating system interface?
8.4 What is meant by preemptive scheduling?
8.5 Which method of process scheduling is most useful in a timesharing environment?
8.6 Describe the steps involved in performing a context switch.
8.7 Describe the programming language hierarchy. Why is a triangle a suitable symbol for representing this hierarchy?
8.8 How does absolute code differ from relocatable code?
8.9 What is a link editor?
8.10 Describe the purpose of each phase of a compiler.
8.11 How does an interpreter differ from a compiler?
8.12 What is a race condition?

Chapter 9 Alternative Architectures
9.1 Why is a RISC processor easier to pipeline than a CISC processor?
9.2 Flynn's taxonomy classifies computer architectures based on two properties. What are they?
9.3 Do all programming problems lend themselves to parallel execution? What is the limiting factor?
9.4 Define superpipelining.
9.5 Give two reasons for the efficiency of vector processors.

Chapter 12 Network Organization and Architecture

12.1 Name and describe the layers of the ISO/OSI Reference Model.
12.2 How is a Network layer protocol different from a Transport layer protocol?
12.3 Which layer of the ISO/OSI Reference Model takes care of negotiating frame size and transmission speed?
12.4 If a communication session were to employ encryption or compression, which layer of the ISO/OSI Reference Model would perform this service?
12.5 Explain the general purpose of the TCP/IP protocol.
12.6 How does IPv6 improve upon IPv4?
12.7 What is the difference between guided and unguided data transmission media? List some examples of each.
12.8 What determines the quality of a transmission medium? What metric is used?
12.9 Where does one find a MAC address? How many bytes are in a MAC address?
12.10 Briefly describe how repeaters, hubs, switches and routers differ from one another.
12.11 When is it not a good idea to use static routing?
12.12 How does link state routing differ from distance vector routing?
12.13 Name some problems that arise from distance vector routing?