* \_\_\_\_\_\_\_ is an arithmetic operation.
* The exclusive OR
* The unary NOT
* **Subtraction**
* The binary AND
* \_\_\_\_\_\_\_ is a logical bit operator.
* The exclusive OR
* The unary NOT
* The binary AND
* **exclusive OR, unary NOT, or binary AND**
* The \_\_\_\_\_\_\_ method of integer representation is the most common method for storing integers in computer memory.
* sign-and-magnitude
* one’s complement
* **two’s complement**
* unsigned integers
* In two’s complement addition, if there is a final carry after the left most column addition, \_\_\_\_\_\_\_.
* **add it to the right most column**
* add it to the left most column
* discard it
* increase the bit length
* For an 8-bit allocation, the smallest decimal number that can be represented in two’s complement form is \_\_\_\_\_\_\_.
* 8
* 127
* **128**
* 256
* For an 8-bit allocation, the largest decimal number that can be represented in two’s complement form is \_\_\_\_\_\_\_.
* 8
* **127**
* 128
* 256
* In two’s complement representation with a 4-bit allocation, we get \_\_\_\_\_\_\_ when we add 1 to 7.
* 8
* 1
* 7
* **8**
* In two’s complement representation with a 4-bit allocation, we get \_\_\_\_\_\_\_ when we add 5 to 5.
* **5**
* 6
* 7
* 10
* If the exponent in Excess\_127 is binary 10000101, the exponent in decimal is \_\_\_\_\_\_\_.
* **6**
* 7
* 8
* 9
* If we are adding two numbers, one of which has an exponent value of 7 and the other an exponent value of 9, we need to shift the decimal point of the smaller number \_\_\_\_\_\_\_.
* one place to the left
* one place to the right
* **two places to the left**
* two places to the right
* \_\_\_\_\_\_\_ operator (s) takes two inputs to produce one output.
* Only AND
* Only OR
* Only XOR
* **AND, OR, or XOR**
* The unary \_\_\_\_\_\_\_ operator inverts its single input.
* AND
* OR
* **NOT**
* XOR
* \_\_\_\_\_\_\_ operator (s), if the input is two 0s, the output is 0.
* In only AND
* In only OR
* In only XOR
* **In AND, OR, or XOR**
* \_\_\_\_\_\_\_ operator (s), if the input is two 1s, the output is 0.
* In only AND
* In only OR
* **In only XOR**
* In AND, OR, or XOR
* For the binary AND operation, only an input of \_\_\_\_\_\_\_ gives an output of 1.
* two 0s
* **two 1s**
* one 0 and one 1
* two 2s
* For the binary OR operation, only an input of \_\_\_\_\_\_\_ gives an output of 0.
* **two 0s**
* two 1s
* one 0 and one 1
* two 2s
* We use a bit pattern called a \_\_\_\_\_\_\_ to modify another bit pattern.
* **mask**
* carry
* float
* byte
* To flip all the bits of a bit pattern, make a mask of all 1s and then \_\_\_\_\_\_\_ the bit pattern and the mask.
* AND
* OR
* **XOR**
* NOT
* To un-set (force to 0) all the bits of a bit pattern, make a mask of all 0s and then \_\_\_\_\_\_\_ the bit pattern and the mask.
* **AND**
* OR
* XOR
* NOT
* To set (force to 1) all the bits of a bit pattern, make a mask of all 1s and then \_\_\_\_\_\_\_ the bit pattern and the mask.
* AND
* **OR**
* XOR
* NOT
* The \_\_\_\_\_\_\_ is a computer subsystem that performs operations on data.
* **CPU**
* memory
* I/O hardware
* bus subsystem
* \_\_\_\_\_\_\_ is a stand-alone storage location that holds data temporarily.
* An ALU
* **A register**
* A control unit
* A tape drive
* \_\_\_\_\_\_\_ is a unit that can add two inputs.
* **An ALU**
* A register
* A control unit
* A tape drive
* A register in a CPU can hold \_\_\_\_\_\_\_.
* only data
* only instructions
* only program counter values
* **data, instruction, or program counter values**
* A control unit with five wires can define up to \_\_\_\_\_\_\_ operations.
* 5
* 10
* 16
* **32**
* A word can be \_\_\_\_\_\_\_ bits.
* only 8
* only 16
* only 32
* **8, or 16, or 32**
* If the memory address space is 16 MB and the word size is 8 bits, then \_\_\_\_\_\_\_ bits are needed to access each word.
* 8
* 16
* **24**
* 32
* The data in \_\_\_\_\_\_\_ is erased if the computer is powered down.
* **RAM**
* ROM
* a tape drive
* a CD-ROM
* \_\_\_\_\_\_\_ is a memory type with capacitors that need to be refreshed periodically.
* SRAM
* **DRAM**
* ROM
* CROM
* \_\_\_\_\_\_\_ is a memory type with traditional flip-flop gates to hold data.
* **SRAM**
* DRAM
* ROM
* CROM
* There are \_\_\_\_\_\_\_ bytes in 16 Terabytes.
* 216
* 240
* **244**
* 256
* \_\_\_\_\_\_\_ can be programmed and erased using electronic impulses but can remain in a computer during erasure.
* ROM
* PROM
* EPROM
* **EEPROM**
* \_\_\_\_\_\_\_ is a type of memory in which the user, not the manufacturer, stores programs that cannot be overwritten.
* ROM
* **PROM**
* EPROM
* EEPROM
* Main memory in a computer usually consists of large amounts of \_\_\_\_\_\_ speed memory.
* high
* medium
* low
* **very high speed**
* A \_\_\_\_\_\_\_ is a storage device to which the user can write information only once.
* CD-ROM
* **CD-R**
* CD-RW
* CD-RR
* A \_\_\_\_\_\_\_ is a storage device that can undergo multiple writes and erasures.
* CD-ROM
* CD-R
* **CD-RW**
* CD-RR
* The smallest storage area on a magnetic disk that can be accessed at one time is a \_\_\_\_\_\_\_.
* track
* **sector**
* frame
* head
* If the memory has 232 words, the address bus needs to have \_\_\_\_\_\_\_ wires.
* 8
* 16
* **32**
* 64
* A control bus with eight wires can define \_\_\_\_\_\_\_ operations.
* 8
* 16
* **256**
* 512
* A \_\_\_\_\_\_\_ controller is a high-speed serial interface that transfers data in packets.
* SCSI
* USB
* **FireWire**
* USB and FireWire
* The three steps in the running of a program on a computer are performed in the specific order \_\_\_\_\_\_\_.
* fetch, execute, and decode
* decode, execute, and fetch
* **fetch, decode, and execute**
* decode, fetch, and execute
* In the \_\_\_\_\_\_\_ method for synchronizing the operation of the CPU with an I/O device, the I/O device informs the CPU when it is ready for data transfer.
* programmed I/O
* **interrupt-driven I/O**
* DMA
* isolated I/O
* In the \_\_\_\_\_\_\_ method for synchronizing the operation of the CPU with an I/O device, the CPU is idle until the I/O operation is finished.
* **programmed I/O**
* interrupt-driven I/O
* DMA
* isolated I/O
* In the \_\_\_\_\_\_\_ method for synchronizing the operation of the CPU with an I/O device, a large block of data can be passed from an I/O device to memory directly.
* programmed I/O
* interrupt-driven I/O
* **DMA**
* isolated I/O

Quizzes: Chapter 06

* The TCP/IP model has \_\_\_\_\_\_\_ layers.
* five
* six
* **seven**
* eight
* The \_\_\_\_\_\_\_ layer of the TCP/IP protocol suite provides services for end users.
* data-link
* transport
* **application**
* physical
* The \_\_\_\_\_\_\_ layer of the TCP/IP protocol suite transmits a bit stream over a physical medium.
* **physical**
* data-link
* network
* The \_\_\_\_\_\_\_ layer of the TCP/IP protocol suite is responsible for node-to-node delivery of a frame between two adjacent nodes.
* transport
* **network**
* data-link
* session
* The \_\_\_\_\_\_\_ layer of the TCP/IP protocol suite is responsible for source-to-destination delivery of the entire message.
* **transport**
* network
* data-link
* session
* What is the domain name in the e‑mail address *kayla@nasa.gov*?
* kayla
* kayla@nasa.gov
* **nasa.gov**
* gov
* Which physical topology uses a hub or switch?
* bus
* ring
* **star**
* bus and ring
* IP addresses are currently \_\_\_\_\_\_\_ bits in length.
* 4
* 8
* **32**
* 40
* \_\_\_\_\_\_\_ protocol (s) is one of the protocols in the transport layer.
* Only TCP
* Only UDP
* Only SCTP
* **TCP, UDP, and SCTP**
* \_\_\_\_\_\_\_ is a protocol for file transfer.
* **FTP**
* SMTP
* TELNET
* HTTP
* \_\_\_\_\_\_\_ is a protocol for e‑mail services.
* FTP
* **SMTP**
* TELNET
* HTTP
* \_\_\_\_\_\_\_ is a protocol for accessing and transferring documents on the WWW.
* FTP
* SMTP
* TELNET
* **HTTP**
* \_\_\_\_\_\_\_ is a program that facilitates the execution of other programs.
* **An operating system**
* Hardware
* A queue
* An application program
* \_\_\_\_\_\_\_ supervises the activity of each component in a computer system.
* **An operating system**
* Hardware
* A queue
* An application program
* Multi-programming requires a \_\_\_\_\_\_\_ operating-system.
* batch
* **time-sharing**
* parallel
* distributed
* \_\_\_\_\_\_\_ is multi-programming with swapping.
* Partitioning
* **Paging**
* Demand paging
* Queuing
* \_\_\_\_\_\_\_ is multi-programming without swapping.
* **Partitioning**
* Virtual memory
* Demand paging
* Queuing
* In \_\_\_\_\_\_\_, only one program can reside in memory for execution.
* **mono-programming**
* multi-programming
* partitioning
* paging
* \_\_\_\_\_\_\_ is a multi-programming method in which multiple programs are entirely in memory with each program occupying a contiguous space.
* **Partitioning**
* Paging
* Demand paging
* Demand segmentation
* In paging, a program is divided into equally sized sections called \_\_\_\_\_\_\_.
* **pages**
* frames
* segments
* partitions
* In \_\_\_\_\_\_\_, the program can be divided into differently sized sections.
* partitioning
* paging
* demand paging
* **demand segmentation**
* In \_\_\_\_\_\_\_, the program can be divided into equally sized sections called pages, but the pages need not be in memory at the same time for execution.
* partitioning
* **paging**
* demand paging
* demand segmentation
* A process in the \_\_\_\_\_\_\_ state can go to either the ready, terminated, or waiting states.
* **hold**
* virtual
* running
* hold or running
* A process in the ready state goes to the running state when \_\_\_\_\_\_\_.
* it enters memory
* it requests I/O
* **it gets access to the CPU**
* it finishes running
* A program becomes a \_\_\_\_\_\_\_ when it is selected by the operating system and brought to the hold state.
* **job**
* process
* deadlock
* partition
* Every process is \_\_\_\_\_\_\_.
* only a job
* only a program
* only a partition
* **a job and a program**
* The \_\_\_\_\_\_\_ scheduler creates a process from a job and changes a process back to a job.
* **job**
* process
* virtual
* queue
* The \_\_\_\_\_\_\_ scheduler moves a process from one process state to another.
* job
* **process**
* virtual
* queue
* To prevent \_\_\_\_\_\_\_, an operating system can put resource restrictions on processes.
* starvation
* synchronization
* paging
* **deadlock**
* \_\_\_\_\_\_\_ can occur if a process has too many resource restrictions.
* **Starvation**
* Synchronization
* Paging
* Deadlock
* The \_\_\_\_\_\_\_ manager is responsible for archiving and backup.
* memory
* process
* device
* **file**
* The \_\_\_\_\_\_\_ manager is responsible for access to I/O devices.
* memory
* process
* **device**
* file