

# Review of essential terms and concepts essay



**ASSIGN  
BUSTER**

Review of Essential Terms and Concepts CIS-300

1. The word bit is a contraction for what two words? Binary digit
2. Explain how the terms bit, byte, nibble, and word are related. A bit is one digit, a nibble consists of four bits, and a byte consists of eight bits, while a word consists of two or more adjacent bits.
3. Why are binary and decimal called positional numbering systems? Each position in a number is weighted by a power of the radix
4. Explain how base 2, base 8, and base 16 are related.

Base 2 each bit stands on its own example 1011011111  
Base 8 you divide 1011011111 into groups of 3 Example 001-011-011-111  
Base 16 you divide 1011011111 into groups of 4 Example 0010-1101-1111

5. What is a radix? In mathematical number systems, the base radix for the simplest case is the number of unique digits, including zero, that are processed in a number system used to represent numbers
6. How many of the “ numbers to remember” (in all bases) from Table 2. 1 can you remember? 10
7. What does overflow mean in the context of unsigned numbers?

This happens when the results of an arithmetic operation is outside the range of the allowable precision for a given number of bits.

8. Name the three ways in which signed integers can be represented in digital computers and explain the differences. A) signed magnitude. B) signed-1's complement. C) signed-2's complement. D). The major difference is the way the decimal place is stored.
9. Which one of the three representations for signed integers is used most often by digital computer systems? Signed magnitude
10. How are complement systems similar to the odometer on a bicycle?

You are essentially cutting system of numbers in half, 0-500 represent positive numbers while 501-999 represents negative numbers, thus making it easier to figure out if you have a positive or negative number. 11. Do you think that double-dabble is an easier method than the other binary-to-decimal conversion methods explained in this chapter? Why? Double-dabble is easier because it is nothing more than simple addition and multiplication. 12. With reference to the previous question, what are the drawbacks of the other two conversion methods?

Using the double-dabble method does not require remembering several different rules such as what signed to use and which number to add or subtract from, it is straightforward math. 13. What is overflow and how can it be detected? How does overflow in unsigned numbers differ from overflow in signed numbers? Overflow is when you have a carryover of a number, thus giving the wrong sign example causing two numbers to be added together, giving the result a negative number. It can be detected by simply knowing if the carry into signed bit equals the carry out signed bit, no overflow has occurred in unsigned numbers.

We know they are positive so overflow does not change this, it will always remain positive, while in signed numbers. This overflow, unless programmed correctly, could change the entire outcome of the intended result. 14. If a computer is capable only of manipulating and storing integers, what difficulties present themselves? How are these difficulties overcome? Should the programmer of the system not take appropriate measures to check all be code written for a specific program, mathematical calculations could produce in accurate results.

Difficulties like these be overcome by checking all programs and having other programmers and users test the system prior to deployment. 15. What are the goals of Booth's algorithm? To increase performance while multiplying twos complement numbers, we have consecutive zeros or ones in the multiplier. 16. How does carry differ from overflow? In carry, the program allows for carryover and will discard carryover bit for the purpose of the answer. It overflow. This number can change the outcome of the problem. Changing a positive number to a negative. This happens when the results do not fit the capacity of storage area. 7. What is arithmetic shifting? An arithmetic shift is a way of multiplying or dividing numbers by two. 18. What are the three component parts of a floating- point number? Signed bit, exponent part, and fractional part (Significand). 19. What is a biased exponent, and what efficiencies can it provide? All integers in a range are converted into nonnegative integers. By using this method no overflow will occur. 20. What is normalization and why is it necessary? Normalization is a database design scheme that prevents duplicate and redundant data. It is necessary to ensure that all data is updated. 1. Why is there always some degree of error in floating- point arithmetic when performed by a binary digital computer? Calculations on a computer are in base two, where the results are in base 10, this difference will always give some degree of error. Depending on how far we carry out the calculation determines the actual error. 22. How many bits long is a double- precision number under the IEEE-754 floating-point standard? 64 23. What is EBCDIC, and how is it related to BCD? Extended Binary Coded Decimal Interchange Code (EBCDIC), it is related to BCD in that the EBCDIC is just an 24.

What is ASCII and how did it originate? American Standard Code for Information Interchanges (ASCII), it originated from the 5-bit code Baudot code which limitations were exceeded so the ASCII instituted the 7-bit coding scheme.

25. Explain the difference between ASCII and Unicode. ASCII is a seven-bit encoding technique which assigns a number to each of the 128 characters used most frequently in American English. Unicode is made up of the 128 characters from the ASCII plus every written alphabet in existence.

26. How many bits does a Unicode character require? 6

27. Why was Unicode created?

28. How do cyclic redundancy checks work? ASCII is a seven-bit encoding technique which assigns a number to each of the 128 characters used most frequently in American English. Unicode is made up of the 128 characters from the ASCII plus every written alphabet in existence.

29. What is systematic error detection? When data is transferred from one device to another certain bits of information are checked to make sure that the data being transferred is accurate. It will not fix the error only report the error.

30. What is a Hamming code? Hamming code is a set of error-correction codes that can be used to detect and correct bit errors that can occur when computer data is moved or stored.

31. What is meant by Hamming distance and why is it important? What is meant by minimum Hamming distance? Hamming distance is the number of bit positions in which two code words differ. To keep data integrity from one device to the other, we need it for error detection. The smallest hamming distance found among all pairs of the code words in that code.

32.

How is the number of redundant bits necessary for code related to the number of data bits? Redundant bits are added into the data bits being

transferred from one source to another, those redundant bits are checked by the receiving unit for accuracy, once the accuracy of that portion of data bits is verified the redundant bits are removed by the receiving unit. 33. What is a burst error? When multiple adjacent bits become damaged a burst error occurs. Usually happens on removable disk storage devices. 34. Name an error detection method that can compensate for burst errors. Reed-Solomon (RS)