Unit 1 Lesson 3 Writing Assignment

**How big of a number can be stored in a byte?**

One byte can be represented by 8 bits so therefore you can store 8 total numbers in a byte that we read from left to right that make up the total sum of 256 because we don’t forget about zero and the range goes from 0 to 255.

**If you add 2 Bytes, what happens to numbers too big to fit?**

When we add two bytes together and the total number exceeds the maximum value we can have in a byte and it causes a byte overflow. When this happens the value wraps around and is then represented by the remainder result instead. For example if we add a byte of 220 with 100 the sum would be 320 which is greater than 256. So, the overflow happens and the result will be 320 minus 256 which would give us 64.

**Explore how negative numbers are represented in binary, ex 12**

When we are talking about negative numbers in binary, the easiest way is sign magnitude although it causes a lot of mistakes. With this method we can figure out if it's a positive or negative value by using the very left number which is commonly known as the most significant bit. So if we look at the first number and its a 0 it represents a positive number and if its a 1, it represents a negative number, for example if we use 12, it can be written as 01100 for the positive 12 and so if we change the significant bit it would be 11100 to represent -12. We can also use a method known as 1’s complement in which we invert all the bits from their original values, so let's use our number 12 again, so we’ll start off by writing it out in positive form in a 8 bit format, so 12 would be 00001100, so if we would invert all the 0’s and 1’s we would get 11110011. However, the most commonly used method is 2’s complement in which it's very similar to 1’s complement but we just add a 1 to the end of the bit, so 11110011 and if we add a 1 it would make 11110100 to represent -12 in 2’s complement.

Louis. E. FrenzelJr (2018). *Electronic Circuits: Digital*. ScienceDirect. https://www.sciencedirect.com/topics/engineering/binary-word#:~:text=The%20maximum%20decimal%20number%20that,overcome%20by%20using%20larger%20words.&text=With%208%20bits%2C%20the%20maximum,256%20or%200%20through%20255.

*Addition of byte values in Java*. Educative. https://www.educative.io/answers/addition-of-byte-values-in-java

Chadwick, R. (n.d.). *Binary Negative Numbers!*. Ryans Tutorials. https://ryanstutorials.net/binary-tutorial/binary-negative-numbers.php