Chapter 7 Review Questions

Explain what AND gate is, give an example.
The AND gate only results in a True output if both of the inputs are True. The truth table for the AND gate is inputs True and True result in a True output, inputs True and False result in a False output, inputs False and True result in a False output, and inputs False and False result in a False output.

Explain what OR gate is, give an example.
The OR gate only gives an output of false if the two inputs are false. The truth table for the OR gate is input True and True results in a True output, input True and False results in a True output, input False and True result in a True output, and input False and False gives a False output.

Explain what XOR gate is, give an example.
The XOR gate means exclusive or. The truth table for the XOR gate is True and True input result in a False output, False and False input result in a False output, True and False input result in a True output, and finally False and True result in a True output. This means that for the output to be True the input can only be exclusively one True and one False for it to result True output otherwise it will result in a false output.

Explain what NOT gate is, give an example.
The NOT gate simply means negate so if you were to input True the output would be False and if you were to input False the output would be True.

Explain what De Morgan's laws are.
The Britannica's explanation of De Morgan's laws are: (1) the negation (or contradictory) of a disjunction is equal to the conjunction of the negation of the alternates that is, not (p or q) equals not p and not q; and (2) the negation of a conjunction is equal to the disjunction of the negation of the original conjuncts that is, not (p and q) equals not p or not q. A basic visual representation the electronic engineering version of De Morgan's laws are (A + B)' = A'B' which represents the OR gate and (AB)' = A' + B' which represents the AND gate.

Write a brief bio for the following luminary figures:
Augustus De Morgan:
As stated in the Britannica Augustus De Morgan lived from 1806-1871. He was an English mathematician and logician. He created the De Morgan's Laws from his understanding of logic of math and he has many contributions to the understanding of mathematical logic. He was one of the mathematicians that recognized the purely symbolic nature of algebra and was aware of the possibility of algebras that differ from regular algebra.

George Boole:
George Boole was and English mathematician in the 19^{th} century that created the Boolean logic. George Boole lived from 1815 – 1864. As stated on the Britannica Boole showed how

it was possible to use algebraic symbols and those that represent logical forms to show how the symbols of quantity can be separated from those of operation. Boolean logic is used for many technological advances like telephone switching and electronic computers that use binary digits and logical elements derived from Boolean logic. Something that I found interesting is that he never got a college degree, and he also became a college professor.

Clifford Berry:
I could not find much information of Clifford Berry, but it seems like his contributions overlap with John Atanasoff. He was born in 1918 and died in 1963. He was also an American Mathematician that went to Iowa State College. He helped create the Atanasoff-Berry Computer or ABC.

Claude Shannon:
Claude Shannon was a student at MIT in the 1930s when he discovered the connection between Boolean logic and how it applies to computing. He realized that we could build digital circuits that would use the Boolean functions. During his time people were hypothesizing the idea of using analog circuits in computers. This would be done by using wire voltages to represent the numbers. Other people had been working on the idea of digital circuits, but they had not considered using Boolean logic in their versions.

John Atanasoff:
According to Britannica John Atanasoff was born in 1903 and died in 1995. He was an American physicist that helped develop the Atanasoff-Berry Computer or ABC with the help of Clifford Berry. The Atanasoff-Berry Computer was able to solve differential equations using binary arithmetic and was one of first digital computers. Atanasoff had been interested in using the power of computers to solve partial differential equations but thought that the analog machines at the time were insufficient. That is why he and Berry decided to work on their own. Apparently, there is some controversy around the ENIAC because Atanasoff had claimed that he and Berry had conversations with Mauchly about the ABC before the ENIAC was created. This led to the creation of the ENIAC.

John Mauchly:
As stated on the Britannica John Mauchly lived from 1907 – 1980 and was an American physicist and engineer. In 1946 alongside John P. Eckert, they created the Electronic Numerical Integrator and Computer or ENIAC which was the first general-purpose electronic computer. Mauchly and Eckert created the ENIAC for the U.S. Army record data for ballistics tests. In 1949 they updated their design created the Binary Automatic Computer or BINAC which used magnetic tape instead of punched cards. Even with some of the ENIAC's shortcomings it was the most powerful calculating device at the time and was the first programmable general-purpose electronic digital computer.

John von Neumann:

John von Neumann was an accomplished physicist and was among the first that recognized how important computation was in scientific research. He helped contribute to the basic architecture of a computer that was designed all the way back in the 1950s. In terms of application programming Neumann recognized that sorting is an essential ingredient in many applications, quadratic-time algorithms are too slow for practical purposes, a divide-and-conquer approach is effective, and proving programs correct and knowing their cost is important. Even with all of the improvements to technology these concepts are still incredibly valuable today and should still be followed.