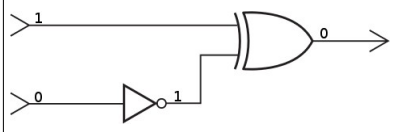
Sheet 3 - Sol

I Solve the following *Review Problems* from *Computer Science: An Overview*:

- **1.1**

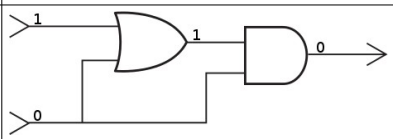
Build the truth table by trying all possible combinations of input and computing the corresponding output.

a)



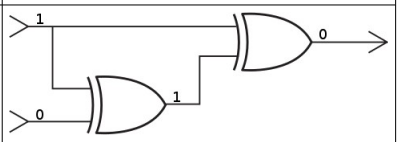
Input	1 1	1 0	0 1	0 0
Output	1	0	0	1

b)



Input	1 1	1 0	0 1	0 0
Output	1	0	1	0

c)



Input	1 1	1 0	0 1	0 0
Output	1	0	1	0

- **1.2**

Build the truth table in the same way like the previous question.

a)

From the truth table:

Input	1 1	1 0	0 1	0 0
Output	1	0	0	0

the logical circuit computes **AND** operation.

b)

From the truth table:

Input	1 1	1 0	0 1	0 0
Output	0	1	1	0

the logical circuit computes **XOR** operation.

- **1.6**

Number of possible values that can be written in base `B` using `n` digits = B^n

For example, using 4 digits in base 10, we can write 10^4=10,000 values (0000 to 9999)

Using two hexadecimal digits: 16^2=256 cells

Using four hexadecimal digits: 16^4=65536 cells

- **1.8**

Convert the given numbers from hexadecimal notations to binary system by replacing every hexadecimal digit with its equivalent four bits and identify the most significant bit.

Actually, it is enough to convert only the most significant hexadecimal digit.

a)	b)	c)	d)
8F=**1**000 1111	FF=**1**111 1111	6F=**0**110 1111	1F=**0**001 1111

- **1.9**

Combine every four bits into the equivalent hexadecimal digit.

a)	b)	c)
1010 0000 1010=A0A	1100 0111 1011=C7B	0000 1011 1110=0BE

II Answer the following questions:

1.

Mbit/s: Mega Bit Per Second
MB: Mega Byte
GB: Giga Byte
Movie size = 1 GB = 1024 MB = 1024 × 8 Mbit
Download time = 1024 × 8/4 = 1024 × 2 = 2048 Seconds ≈ 34 Minutes

2.

Using only one bit, we can address 2^1 = 2 drawers {0,1}
Using two bits, we can address 2^2 = 4 drawers {00,01,10,11}
Using three bits, we can address 2^3 = 8 drawers {000,001,010,011,100,101,110,111}
Using four bits, we can address 2^4 = 16 drawers {...}
Then we need 4 bits.
Another way to solve the question is to find $\lceil \log_2(12) \rceil = \lceil 3.3 \rceil = 4$

3.

a) *Bit*
Binary Digit
{0,1}

b) *Byte*
1 Byte = 8 Bit
{00000000,00000001,00000010,...,11111111}

c) *Boolean Operation*
Mathematical operation performed on binary digits
{**AND, OR, XOR, NOT**}