Boolean Algebra

1. Simplify the following expressions using Boolean algebraic laws. Give each step of your simplification and denote which laws you're using for each step. Do not skip or combine steps!

a) A * ($\overline{A}$ + B * B) + $\overline{(B+A)}$ * ($\overline{A}$ + B)

= A * ($\overline{A}$ + B) + $\overline{(B+A)}$ * ($\overline{A}$ + B) ———Idempotent

= A$\overline{A}$ + AB + $\overline{(B+A)}$ * ($\overline{A}$ + B) ———distributive

= AB + $\overline{(B+A)}$ * ($\overline{A}$ + B) ———complement

= AB + $\overline{B}$ * $\overline{A}$ * ($\overline{A}$ + B) ———deMorgan

= AB + $\overline{B}$ $\overline{A}$ $\overline{A}$ + $\overline{B}$ $\overline{A}$ B ———distribute

= AB + $\overline{B}$ $\overline{A}$ + $\overline{B}$ $\overline{A}$ B ———idempotent

= AB + $\overline{B}$ $\overline{A}$ + 0 ———complement

= AB + $\overline{B}$ $\overline{A}$

b) $\overline{C * B}$ + (A * B * C) + $\overline{A + C + \overline{B}}$

= $\overline{C}$ + $\overline{B}$ + (ABC) + $\overline{A + C + \overline{B}}$ ——— deMorgan

= $\overline{C}$ + $\overline{B}$ + (ABC) + $(\overline{A + C})$ * $\overline{\overline{B}}$ ——— deMorgan

= $\overline{C}$ + $\overline{B}$ + (ABC) + $(\overline{A}$ * $\overline{C})$ * $\overline{\overline{B}}$ ——— deMorgan

= $\overline{C}$ + $\overline{B}$ + (ABC) + $(\overline{A}B\overline{C})$ ——— distributive

= $\overline{C}$ (1+ $\overline{A}B\overline{C}$) + $\overline{B}$ + (ABC) ——— absorption

= $\overline{C}$ + $\overline{B}$ + (ABC) ——— annulment

= $\overline{C}$ + $\overline{B}$ + AB ——— absorption

= $\overline{C}$ + $\overline{B}$ + A ——— absorption

c) (A + B) * ($\overline{A}$ + C) * ($\overline{C}$ + B)

= (A$\overline{A}$ + AC + B$\overline{A}$ + BC) * ($\overline{C}$ + B) ——— distributive

= (AC + B$\overline{A}$ + BC) * ($\overline{C}$ + B) ——— idempotent

= AC$\overline{C}$ + ACB + B$\overline{A}$*$\overline{C}$ + B$\overline{A}$B + BC$\overline{C}$ + BCB ——— distributive

= ACB + B$\overline{A}$*$\overline{C}$ + B$\overline{A}$B + BCB ——— complement

= ACB + B$\overline{A}$*$\overline{C}$ + B$\overline{A}$ + BC ——— idempotent

= ACB + B$\overline{A}$($\overline{C}$ + 1) + BC ——— absorption

= ACB + B$\overline{A}$ + BC ——— annulment

= BC(A+1) + B$\overline{A}$ ——— absorption

= BC + B$\overline{A}$ ——— annulment

2) Find all solutions of the following Boolean equations without using the truth tables:

a) $(\overline{A} + \mathrm{C}) * (\overline{B} + \mathrm{D} + \mathrm{A}) * (\mathrm{D} + \mathrm{A} * \overline{C}) * (\overline{D} + \mathrm{A}) = 1$

$(\overline{A}\,\overline{B} + \overline{A}\mathrm{D} + \overline{A}\mathrm{A} + \overline{B}\mathrm{C} + \mathrm{CD} + \mathrm{AC}) * (\mathrm{D} + \mathrm{A} * \overline{C}) * (\overline{D} + \mathrm{A}) = 1$

$(\overline{A}\,\overline{B} + \overline{A}\mathrm{D} + \overline{B}\mathrm{C} + \mathrm{CD} + \mathrm{AC}) * (\mathrm{D} + \mathrm{A} * \overline{C}) * (\overline{D} + \mathrm{A}) = 1$

$(\overline{A}\,\overline{B} + \overline{A}\mathrm{D} + \mathrm{CD} + \mathrm{AC}) * (\mathrm{D} + \mathrm{A} * \overline{C}) * (\overline{D} + \mathrm{A}) = 1$

$(\overline{A}\,\overline{B} + \overline{A}\mathrm{D} + \mathrm{AC}) * (\mathrm{D} + \mathrm{A} * \overline{C}) * (\overline{D} + \mathrm{A}) = 1$

$(\overline{A}\,\overline{B}\mathrm{D} + \overline{A}\mathrm{DD} + \mathrm{ACD} + \overline{A}\,\overline{B}\mathrm{A}\overline{C} + \overline{A}\mathrm{DA}\overline{C} + \mathrm{ACA}\overline{C}) * (\overline{D} + \mathrm{A}) = 1$

$(\overline{A}\,\overline{B}\mathrm{D} + \overline{A}\mathrm{D} + \mathrm{ACD}) * (\overline{D} + \mathrm{A}) = 1$

$(\overline{A}\mathrm{D}(\overline{B} + 1) + \mathrm{ACD}) * (\overline{D} + \mathrm{A}) = 1$

$(\overline{A}\mathrm{D} + \mathrm{ACD}) * (\overline{D} + \mathrm{A}) = 1$

$(\overline{A}\mathrm{D}\overline{D} + \overline{A}\mathrm{DA} + \mathrm{ACD}\overline{D} + \mathrm{ACDA}) = 1$

$\mathrm{ACDA} = 1$

$\mathrm{ACD} = 1$

Solution: when ACD all equal to 1

b) $(((\overline{K}\mathrm{LN}) * (\mathrm{L} + \mathrm{M})) + ((\overline{K} + \mathrm{L} + \mathrm{N}) * (\mathrm{K}\overline{L}\,\overline{M}))) * (\overline{K} + \overline{N}) = 1$

$[(\overline{K}\mathrm{LNL} + \overline{K}\mathrm{LNM}) + (\overline{K}\mathrm{K}\overline{L}\,\overline{M} + \mathrm{LK}\overline{L}\,\overline{M} + \mathrm{NK}\overline{L}\,\overline{M})] * (\overline{K} + \overline{N}) = 1$

$[(\overline{K}\mathrm{LN} + \overline{K}\mathrm{LNM}) + (\mathrm{NK}\overline{L}\,\overline{M})] * (\overline{K} + \overline{N}) = 1$

$[(\overline{K}\mathrm{LN}\ (1 + \mathrm{M})) + (\mathrm{NK}\overline{L}\,\overline{M})] * (\overline{K} + \overline{N}) = 1$

$(\overline{K}\mathrm{LN} + \mathrm{K}\overline{L}\,\overline{M}\mathrm{N}) * (\overline{K} + \overline{N}) = 1$

$\overline{K}\,\overline{K}\mathrm{LN} + \mathrm{K}\overline{K}\,\overline{L}\,\overline{M}\mathrm{N} + \overline{K}\mathrm{LN}\overline{N} + \mathrm{K}\overline{L}\,\overline{M}\mathrm{N}\overline{N} = 1$

$\overline{K}\mathrm{LN} = 1$

Solution: when K is 0, and L,N = 1

3) Simplify the following expression by first constructing a truth table, using that truth table to construct a K-map, and then using that K-map to simplify.

$$Q = \overline{X} * \overline{Y} * Z + X * Y * \overline{Z} + \overline{X} * Y * \overline{Z} + X * \overline{Y} * \overline{Z}$$

X	Y	Z	Q
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

Z \ XY	00	01	11	10
0	0	1	1	1
1	1	0	0	0

$$Q = \overline{X}\,\overline{Y}Z + Y\overline{Z} + X\overline{Z}$$

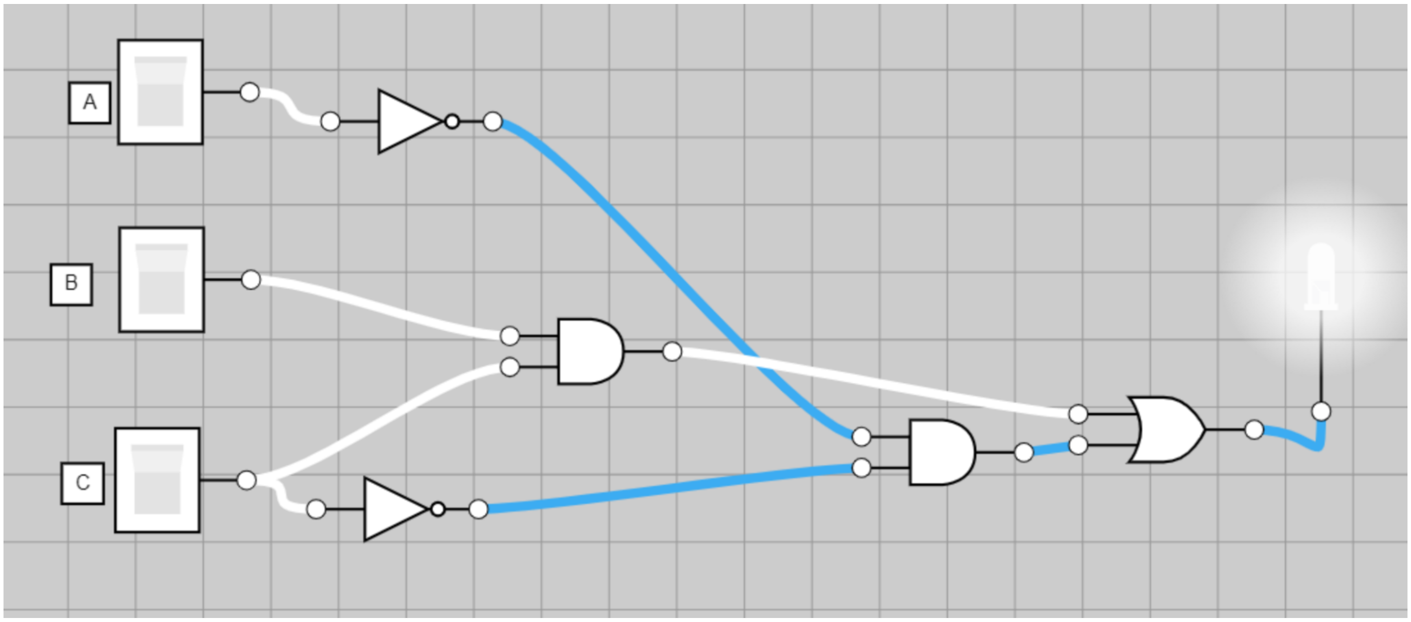
4) Convert the following truth table into its sum of products representation:

A	B	C	Output
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

C \ AB	00	01	11	10
0	1	1	0	0
1	0	1	1	0

$$\text{OUTPUT} = \overline{A}\,\overline{C} + \text{BC}$$

5) Draw a logical circuit diagram that represents the above sum of products expression using OpenCircuits (https://opencircuits.io/). Clearly label all inputs/outputs and all components. Make sure you connect appropriate input components (e.g., buttons, switches, clocks, etc.) and output components (e.g., LEDs, displays, etc.) to facilitate testing of your circuit. Download your diagram using OpenCircuits' "Download" feature, rename it to hw4_SOP.circuit, and submit on Submitty along with your hw4.pdf file.

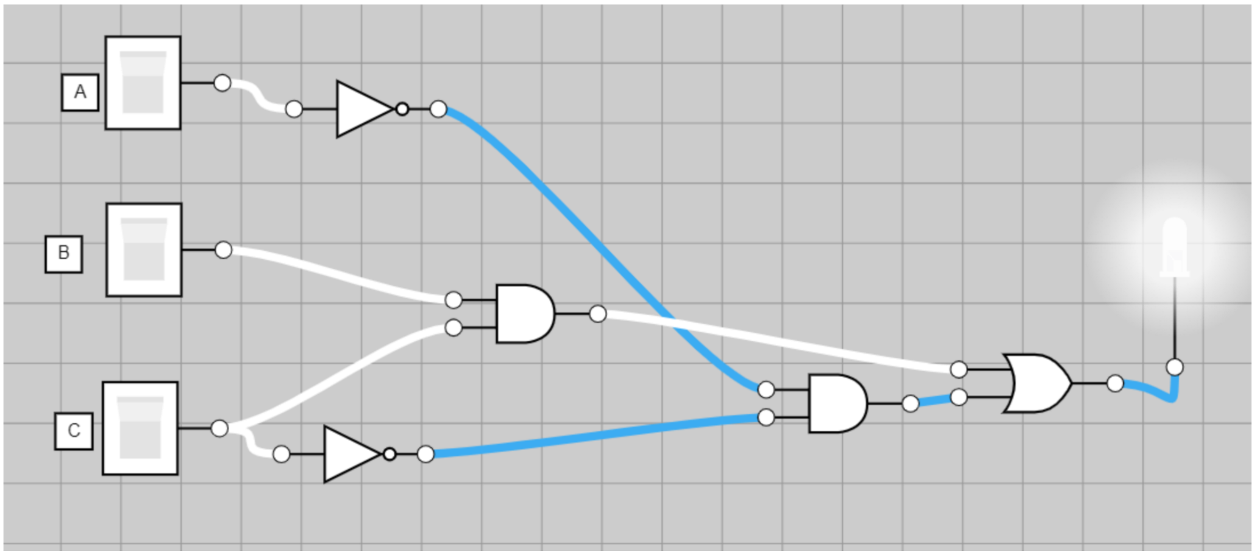


6) Test you circuit by supplying appropriate inputs and observing the expected values of the output. Explain why your set of tests is sufficient to prove that your logical circuit does in fact implement the required Boolean function. For each test, provide a picture (snapshot) of your circuit. Insert all such pictures in the hw4.pdf PDF file. You can download pictures (PNG, JPEG, or PDF) of your circuit diagram using OpenCircuits' "Export Image" feature.

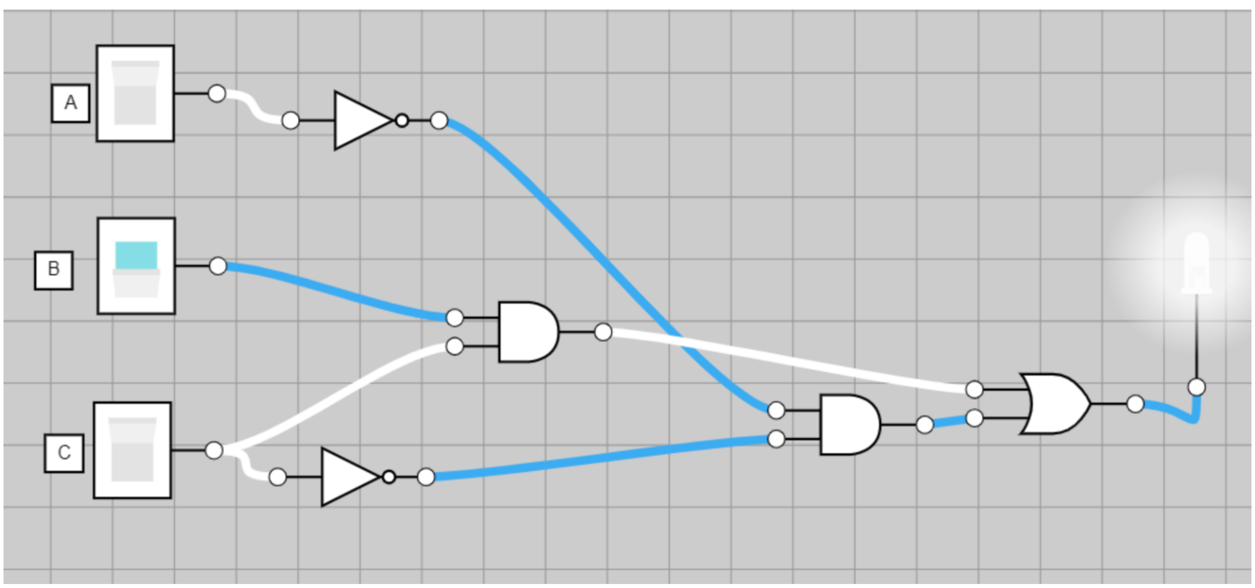
According to our truth table, the led will be on when:

A = 0, B = 0, C = 0
A = 0, B = 1, C = 0
A = 0, B = 1, C = 1
A = 1, B = 1, C = 1

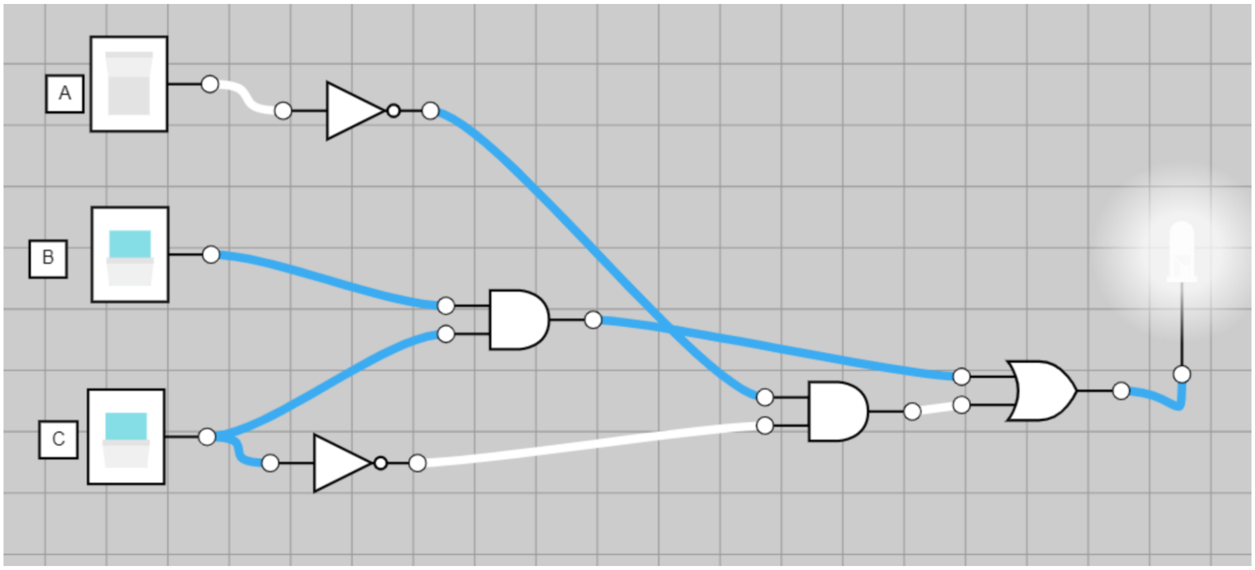
A = 0, B = 0, C = 0



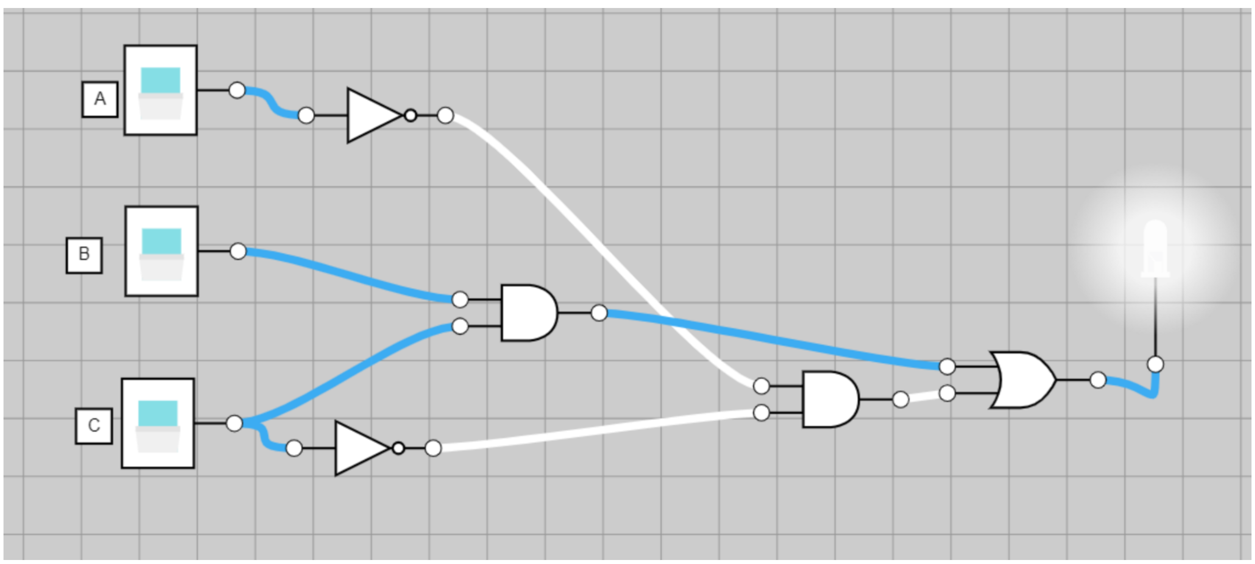
A = 0, B = 1, C = 0



A = 0, B = 1, C = 1



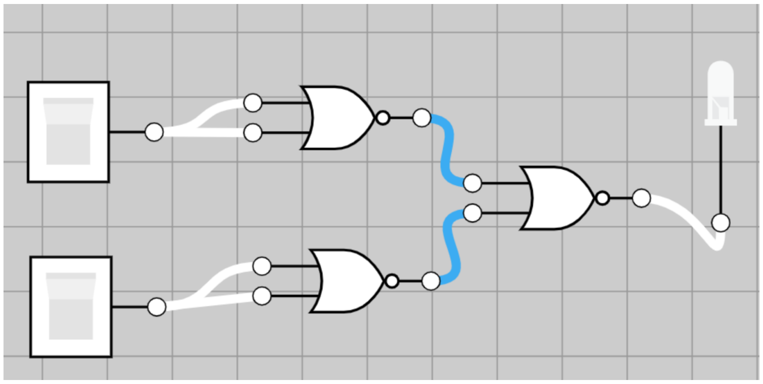
A = 1, B = 1, C = 1



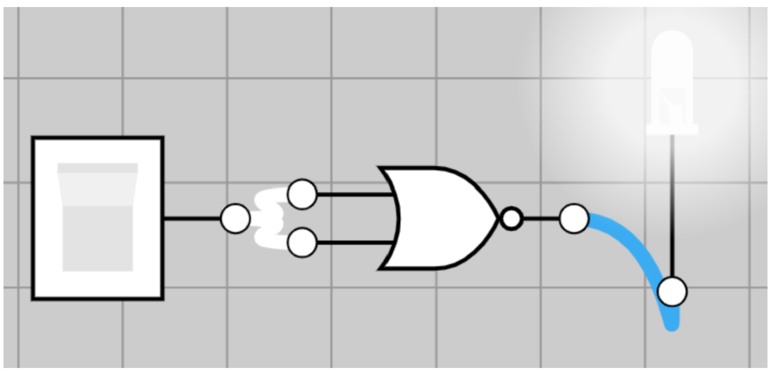
In all these cases, the led is on, meaning output is 1

7)

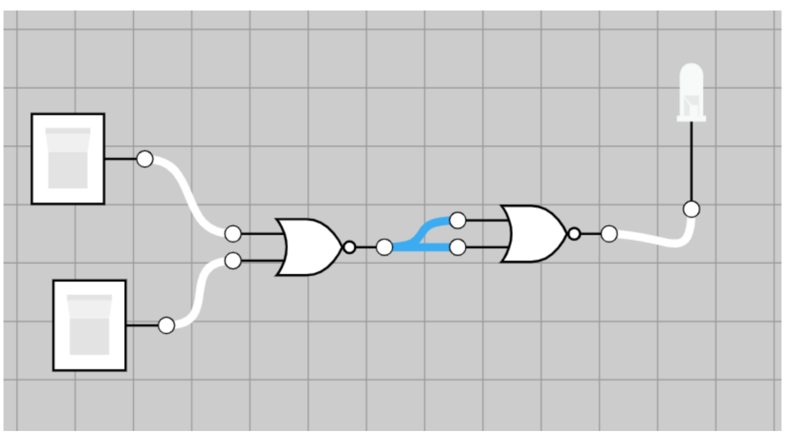
AND



NOT



OR



<u>Textbook Problem 9.13.1</u>
5ED4 in binary:
0101 1110 1101 0100
07A4 in binary:
0000 0111 1010 0100

0101 1110 1101 0100
-0000 0111 0011 0100
—-------------------------
0101 0111 0011 0000
= 5730 in hexadecimal

<u>Textbook Problem 9.13.2</u>
Will also be 5730, because the sign bit is 0

<u>Textbook Problem 9.13.6</u>
185 in binary:
10111001
122 in binary:
01111010

10111001
-01111010
—-------------
00111110
Neither overflow or underflow

<u>Textbook Problem 9.13.10</u>
151 in binary:
10010111
214 in binary:
11010110

151 in 2's complement:
01101000 + 1 = 01101001 (-105)
214 in 2's complement:
00101001 + 1 = 00101010 -42)
(-105) - (-42) = **-63**

<u>Textbook Problem 9.13.11</u>
151 in binary:
10010111
214 in binary:
11010110

 10010111
+11010110
—-------------
1 01101101

Exceeds max, thus, 151 + 214 = 1111 1111 = 255

Textbook Problem 9.13.20
0 x 0C000000 in binary:
0000 1100 0000 0000 0000 0000 0000 0000

in decimal:
1 x 2E27 + 1 x 2E26 = 201326592

un-signed is the same because the the leading bit is a 0

Textbook Problem 9.13.21
Mips instruction: opcode (00011) is jal or jal 0

15) Give a reason why we use two's complement representation for negative numbers in computer arithmetic. Give an example of its usage.

when adding positive and negative numbers, it's more intuitive to use two's complement representation because we won't need to apply a different logic.

for example: let's say we want to add 5 and -1
using 2's complement approach:
5 = 0101
-1 = 1111
 0101
+1111
—-------
0100 = 4, which is correct

using 1st left bit as the sign approach:
5 = 0101
-1 = 1001
 0101
+1001
—--------
1110 = -6, which is definitely not right