Assigned reading: BF Sections 1 and 2

(1) (BF 1.2) Let \( r = \text{“she registered to vote”} \) and \( v = \text{“she voted”} \). Write the following statement in symbolic form: She registered to vote but she did not vote.

\[ r \land \neg v. \]

In English, “but” is “and” with an underlying message: The use of “but” in this way often (but not always) indicates surprise. You might say, “The animal is a fish \text{and} it lives in the water.” \((F \land W)\) Or you might say “The animal is a fish \text{but} it lives on dry land.” \((F \land L)\) Logic doesn’t make a fuss over surprises.

(2) (BF 1.5) Make a truth table for \((p \lor (\neg p \lor q)) \land \neg (q \land \neg r)\)

Before making a truth table, it may help to simplify the expression. Using the associative law \( p \lor (\neg p \lor q) = (p \lor \neg p) \lor q = 1 \lor q = 1. \) Thus

\[ (p \lor (\neg p \lor q)) \land \neg (q \land \neg r) = 1 \land \neg (q \land \neg r) = \neg (q \land \neg r) = \neg q \lor r, \]

where the last is by DeMorgan’s Law. Now we are ready to make the table.

<table>
<thead>
<tr>
<th>( p )</th>
<th>( q )</th>
<th>( r )</th>
<th>( \sim q )</th>
<th>( \sim q \lor r )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

(3) (BF 1.7) Using DeMorgan’s rule, state the negation of the statement: “The car is out of gas or the fuel line is plugged.”

Let \( g = \text{“The car has gas”} \) and let \( f = \text{“The fuel line is plugged.”} \) The statement is \( \sim g \lor f \). Its negation is \( \sim (\sim g \lor f) = g \land \sim f \). In words, “The car has gas and the fuel line isn’t plugged.”

You could have taken \( h = \text{“The car is out of gas”}. \) The statement and its negation would be \( h \lor f \) and \( \sim (h \lor f) = \sim h \land \sim f \). In words, “The car is not out of gas and the fuel line isn’t plugged.” There is a double negative in “is not out of gas,” which you could simplify to “has gas.”

(4) A pair of numbers \( x \) and \( y \) satisfy a system of inequalities if

\[
\begin{align*}
3 & \leq x \leq 5 \quad \text{and} \\
|x - y| & < 1.
\end{align*}
\]

What are the conditions under which \( x \) and \( y \) fail to satisfy this system?

\( x \) and \( y \) satisfy the system of inequalities if and only if both conditions are met. To find the conditions under which they fail, we simply take the negation of the system. This is
the principle of DeMorgan’s law. Therefore $x$ and $y$ will fail to satisfy the system when any part of either conditions is not met: when $x < 3$ or $x > 5$ or $|x - y| \geq 1$.

(5) (BF 1.15) Is the function $(p \land (\sim (\sim (p \lor q)))) \lor (p \land q)$ equal to the function $p \lor q$? Why or why not?

No. You could use a truth table:

<table>
<thead>
<tr>
<th>$p$</th>
<th>$q$</th>
<th>$\sim p$</th>
<th>$\sim (\sim (p \lor q))$</th>
<th>$p \land (\sim (\sim (p \lor q))) \lor (p \land q)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

As you can see, the last two columns are not equivalent and thus the function $(p \land (\sim (\sim (p \lor q))) \lor (p \land q)$ is not equal to the function $p \lor q$. Otherwise, you could use algebraic simplification:

\[
(p \land (\sim (\sim (p \lor q))) \lor (p \land q) = (p \land \sim q) \lor (p \land q) = p \land (\sim q \lor q) = p
\]

$p$ is not equivalent to $p \lor q$.

(6) (BF 2.10) Convert the following as indicated.

(a) Convert 61502$_8$ to decimal.

- $61502_8 = 6 \times 8^4 + 1 \times 8^3 + 5 \times 8^2 + 2 = 25410$.

(b) Convert $EB7C5_{16}$ to octal.

- $EB7C5_{16} = 1110\ 1011\ 0111\ 1100\ 0101_2 = 11\ 101\ 011\ 011\ 111\ 000\ 101_2 = 3533705_8$.

(7) (BF 2.14) The number 10001001$_2$ is the 8-bit two’s complement of a number $k$. What is the decimal representation of $k$?

- First method: Start with 10001001. Using the two’s complement algorithm this converts to 01110111 which is $64 + 32 + 16 + 4 + 2 + 1 = 119$ and so $k = -119$.

- Second method: 10001001$_2 = 128 + 8 + 1 = 137$. The absolute value of its two’s complement in an 8-bit register is $2^8 - 137 = 119$. Thus, it’s representing the number $-119$.

(8) Add 11111111$_2 + 1$ and convert the result to decimal representation. Notice that this verifies that 11111111$_2 = (2^8 - 1)_{10}$. Generalize this observation to make a conjecture about what the decimal representation of \underbrace{111\cdots111}_n$_2$ is.

\[
11111111_2 + \underbrace{1_2}_{n\ times} = 2^n
\]

Notice that our sum has an overflow problem in which the 1 that we carry over is out of place and so we can just drop it! (See Example 14 on BF-15). The decimal representation of \underbrace{111\cdots111}_n$_2$ is $2^n - 1_{10}$.
(9) (BF 2.16) Using base-2 arithmetic, compute \(79 - 43\). Then compute it using 8-bit two’s complement registers. Remember to check for overflow.

We first convert the two numbers into 8-bit binary formats.

\[
79_{10} = (64 + 8 + 4 + 2 + 1)_{10} = 01001111_2
\]

\[
43_{10} = (32 + 8 + 2 + 1)_{10} = 00101011_2
\]

Then since \(79 > 43\), we take binary arithmetic operation on them and get the result \(00100100_2\). In the other way, using two’s complement, we first convert the negative number \(-43\) to be \(11010101_2\) and take the addition operation with \(01001111_2\). Notice that any carry bit is seen as overflow and will be discarded, and after bitwise addition the result turns out to be also \(00100100_2\).

(10) Show that if \(a\) and \(b\) are integers in the range 1 through 128, and the sum of \(a\) and \(b\) is also in this range, then

\[
2^8 \leq (2^8 - a) + (2^8 - b) < 2^9.
\]

*Note:* \(2^8 = 256\). Explain why it follows that the binary representation of \((2^8 - a) + (2^8 - b)\) has a leading 1 in the \(2^8\)th position.

**Theorem:** If \(a\) and \(b\) are integers in the range 1 through 128, and the sum of \(a\) and \(b\) is also in this range, then \(2^8 \leq (2^8 - a) + (2^8 - b) < 2^9\).

**Given:** Assume \(a\) and \(b\) are integers in the range 1 - 128 and their sum is also in this range.

**Want To Show (WTS):** \(2^8 \leq (2^8 - a) + (2^8 - b) < 2^9\)

**Proof:**

We can rearrange the middle of our intended inequality to make it easier to connect to our assumptions.

\[
(2^8 - a) + (2^8 - b) = 2^9 - (a + b)
\]

Now, because the sum of \(a\) and \(b\) must be in the range 1 through 128, we know that the smallest value of \(a + b\) is 1 and the largest is 128. To check if \(2^8 \leq \) the smallest possible value of \(2^9 - (a + b)\), we plug in the largest value of \(a + b\). To check if \(2^9 > \) the largest possible value of \(2^9 - (a + b)\), we plug in the smallest value.

\[
2^8 = 256 < 384 = 2^9 - (128) \leq 2^9 - (a + b)
\]

Stringing these together, we get \(2^8 \leq 2^9 - (a + b)\). For the upper bound:

\[
2^9 - (a + b) \leq 2^9 - 1 < 2^9.
\]

**Conclusion:**

Therefore, if \(a\) and \(b\) are integers in the range 1 through 128, and the sum of \(a\) and \(b\) is also in this range, then \(2^8 \leq (2^8 - a) + (2^8 - b) < 2^9\). The binary representation of \((2^8 - a) + (2^8 - b)\) has a leading 1 in the \(2^8\)th position because the range of values that it represents (all integers between 384 and 511) is always greater than \(2^8\).