Module 1: Propositional Logic
Module 1: Propositional Logic

- By the start of the class, you should be able to:
  - Translate back and forth between simple natural language statements and propositional logic.
  - Evaluate the truth of propositional logic statements using truth tables.
  - Translate back and forth between propositional logic statements and circuits that assess the truth of those statements.
Quiz 1 feedback:

- Very very well done overall.
- Be careful:
  - $\neg a \lor \neg b$ is not the same as $\neg(a \lor b)$.
  - $\neg a \land \neg b$ is not the same as $\neg(a \land b)$.
- We will discuss the open-ended question a bit later.
Module 1: Coming up...

- What is coming up?
  - Second pre-class quiz: due tomorrow September 16\textsuperscript{th} at 19:00.
  - Assigned reading for the quiz:
    - Epp, 5\textsuperscript{th} or 4\textsuperscript{th} edition: 2.2
    - Epp, 3\textsuperscript{rd} edition: 1.2
    - Rosen, 6\textsuperscript{th} or 7\textsuperscript{th} edition: 1.1 from page 6 onwards.
  - Assignment #1
    - due Monday September 28\textsuperscript{th}, 2020 at 19:00.
Module 1: Coming up...

What is coming up?

- Third pre-class quiz: due Wednesday September 23rd at 19:00.
- Assigned reading for the quiz:
  - Epp, 5th edition: 2.5
  - Epp, 4th edition: 2.5 +
    http://www.ugrad.cs.ubc.ca/~cs121/current/handouts/signed-binary-decimal-conversions.html
Module 1: Propositional Logic

- CPSC 121: the **BIG** questions:
  - We are not yet ready to directly address any of the big questions.
  - But this module lays out the groundwork for all of them.
Module 1: Propositional Logic

By the end of this module, you should be able to:

- Build computational systems to solve real problems, using both propositional logic expressions and equivalent digital logic circuits,
  - The light switches problem from the 1st pre-class quiz.
  - The 7- or 4-segment LED displays we will discuss in class.
Module 1: Propositional Logic

- Module outline:
  - Writing a truth table
  - From circuits to propositions
  - Light switches
  - 7-segment displays
  - More exercises
Module 1.1: Writing a truth table

- **Must** the combinations in a truth table be listed in a specific order?

- Our recommendation:
  - Why?
Module 1.1: Writing a truth table

- We will always start with false (you will see why later).

- With 3 variables:
  - the first column contains 4 false followed by 4 true.
  - the second column contains 2 false, 2 true, 2 false, 2 true.
  - and the third column alternates false with true.
Module 1.2: Writing a truth table

- With \( k \) variables:
  - the first column has \( 2^{k-1} \) false and then \( 2^{k-1} \) true
  - the second column has \( 2^{k-2} \) false and then \( 2^{k-2} \) true (twice)
  - etc.
Module 1.2: Writing a truth table

- Another way to get the same list of combinations:
  - Write a truth table with $k-1$ variables twice.
Module 1.2: Writing a truth table

- Another way to get the same list of combinations:
  - Write a truth table with \(k-1\) variables twice.
  - Add false in front of the first copy, and true in front of the second one.

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
</tr>
</tbody>
</table>
Module 1.2: Writing a truth table

- Another way to get the same list of combinations:
  - Write a truth table with \( k-1 \) variables twice.
  - Add \textit{false} in front of the first copy, and \textit{true} in front of the second one.

<table>
<thead>
<tr>
<th>F</th>
<th>F</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
</tbody>
</table>
Module 1: Propositional Logic

Module outline:

- Writing a truth table
- From circuits to propositions
- Light switches
- 7-segment displays
- More exercises
Module 1.2: From circuits to propositions

- We can use propositions to model circuits:
  - Each variable corresponds to an input.
  - The value of the proposition (for a given combination of input values) is the output of the circuit.
  - Example: the segments in a 7 segment display
Example (continued)

- We use ? inputs (variables).
- Each integer from 0 to 9 is represented by a specific combination of input values.
- We can write a proposition that tells us for which combinations of input values the segment should be turned on.
Module 1.2: From circuits to propositions

- We will spend time designing circuits, but first how do we go from a circuit to the proposition that models its output?
- Example: what does this circuit compute?
Module 1.2: From circuits to propositions

- To find the logical expression that corresponds to a circuit's output:
  - First we write the operator for the gate that produces the circuit's output.
  - The operator's left argument is the expression for the circuit connected to the gate's first input.
  - The operator's right argument is the expression for the circuit connected to the gate's second input.
- This is our first algorithm!
Module 1.2: From circuits to propositions

- Example:

- What does this circuit compute?
Module 1.2: From circuits to propositions

What is the simplest logical expression that corresponds to the following circuit?
Module 1: Propositional Logic

- Module outline:
  - Writing a truth table
  - From circuits to propositions
  - Light switches
  - 7-segment displays
  - More exercises
Module 1.3: Light switches

- Consider again:
  - Design a light that changes state whenever any of the switches that control it is flipped. Ideally your solution would work with any number!
How do we approach this?

- We first need to make sure we understand what we are designing.
- Then we use propositional logic to model the circuit’s desired output.
- It’s helpful to start with very simple versions of the problem
  - First try 1 switch.
  - Then try 2 switches.
  - Then try 3 switches.
  - Then see if we can generalize to $n$ switches.
Module 1.3: Light switches

Making sure we understand what we are designing.

Which of these would be usable (most useful) as the output of our circuit?

- a) the switch is flipped
- b) the switch is on
- c) the light is on
- d) the light changed state
Module 1.3: Light switches

- Making sure we understand what we are designing.
  - Which of these would be usable (most useful) as input to our circuit?
    a) the switch is flipped
    b) the switch is on
    c) the light is on
    d) the light changed state
Module 1.3: Light switches

- One switch:
  - make sure we understand the problem first.
  - Is the light on or off when the switch is “on”?
    a) Always on.
    b) Always off.
    c) Depends, but a correct solution should always do the same thing.
    d) Depends, and a correct solution might do different things at different times.
Module 1.3: Light switches

- One switch:
  - Which circuit(s) is/are correct solution(s)?

  a) ![Switch Circuit A]

  b) ![Switch Circuit B]

  c) ![Switch Circuit C]

  d) two of a, b, c.

  e) all three of a, b, c.
Module 1.3: Light switches

- Two switches:
  - make sure we understand the problem first.
  - Is the light on or off when both switches are “on”?
    a) Always on.
    b) Always off.
    c) Depends, but a correct solution should always do the same thing for a given setting of the switches.
    d) Depends, and a correct solution might do different things at different times for the same setting of the switches.
Module 1.3: Light switches

- Two switches: which circuit(s) work(s)?

a) [Diagram]

b) [Diagram]

c) [Diagram]

d) Both a and b work.

e) Both b and c work.
Module 1.3: Light switches

- Circuit design tip: if you are not sure where to start while designing a circuit,
  - First determine the inputs and the output
  - Then build the truth table.
  - Finally turn it into a circuit.

- For the three switches problem:
  - We can decide arbitrarily what the output is when all three switches are OFF.
  - This determines the output for all other cases!
  - Let's see how...
Module 1.3: Light switches

- What pattern do we observe?
  - The light is ON if

- Now to generalize to \( n \) switches...
  - What do you think the answer is?

- How can we convince ourselves that it is correct?
  - Mathematical induction
Module 1: Propositional Logic

- Module outline:
  - Writing a truth table
  - From circuits to propositions
  - Light switches
  - 7-segment displays
  - More exercises
Module 1.3: 7-segment displays

- Problem: design a circuit that displays the numbers 0 through 9 using seven LEDs (lights) in the shape illustrated below.
Module 1.4: 7-segment displays

- How do we represent the inputs?
  - Use ? logical (true/false) values.
  - Each integer represented by 1 specific combination of input values.
- Could we do this arbitrarily?
  - Yes!
- But we won't
  - How many of you know about binary representation?
Module 1.4: 7-segment displays

- Understanding: what is the smallest number of \textbf{inputs} (wires going into the circuit) possible?
  a) 1
  b) 4
  c) 7
  d) 10
  e) None of the above
Module 1.4: 7-segment displays

Here's how we will do it:

<table>
<thead>
<tr>
<th>Number</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>1</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>2</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>3</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>4</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>5</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>6</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>7</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>8</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>9</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>T</td>
</tr>
</tbody>
</table>
Module 1.4: 7-segment displays

- Understanding: how many outputs (lights) are there?
  
a) 1
  
b) 4
  
c) 7
  
d) 10
  
e) None of the above
Module 1.4: 7-segment displays

- Suppose 7 of you were each in charge of one segment.
- Which other person's algorithm would you need to know about?
  
  a) No one else's.
  b) The students in charge of neighbouring segments.
  c) The person in charge of the segment in front of yours.
  d) Everybody else's.
  e) None of the above.
Module 1.4: 7-segment displays

- Let us look at the bottom-left segment. Which is the correct truth table?

<table>
<thead>
<tr>
<th>Number</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>1</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>2</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>3</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>4</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>5</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>6</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>7</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>8</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>9</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>T</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>(a)</th>
<th>(b)</th>
<th>(c)</th>
<th>(d)</th>
<th>(e)</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>F</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>F</td>
<td>4</td>
<td>6</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>T</td>
<td></td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>F</td>
<td></td>
<td></td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>F</td>
<td></td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>F</td>
<td></td>
<td></td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td></td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>F</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Module 1.4: 7-segment displays

- We will discuss two different approaches to write a proposition that describes this truth table:
  - One that’s mechanical and easy to use, but gives very long propositions in general.
  - One that relies on spotting patterns: it’s harder but produces much shorter propositions.
- There is a third approach using something called Karnaugh maps; we will not discuss it in class or expect you to use it on exams.
Module 1.4: 7-segment displays

Which proposition is true only for the red row?

a) \( a \land b \lor c \land d \)

b) \( \neg a \lor \neg b \lor c \lor \neg d \)

c) \( \neg a \land \neg b \land c \land \neg d \)

d) \( a \oplus b \oplus c \oplus d \)

e) None of the above

<table>
<thead>
<tr>
<th>Nb</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>1</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>2</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>3</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>4</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>5</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>6</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>7</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>8</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>9</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>T</td>
</tr>
</tbody>
</table>
Module 1.4: 7-segment displays

Worksheet Questions #1 to #7
Module 1: Propositional Logic

• Module outline:
  • Writing a truth table
  • From circuits to propositions
  • Light switches
  • 7-segment displays
  • More exercises
Module 1.4: More exercises

- Prove that our two solutions for the lower-left segment are not logically equivalent.
  - You should do this by providing values for the variables, so the two propositions have different truth values.
  - Why are they both correct solutions, despite that?
- Finish the problem by building circuits for the other 5 segments.
- Design a circuit that takes three bits as input, and outputs the binary representation for their sum.
Module 1.4: More exercises

- Build a circuit that displays the numbers 1 through 9 represented by four Boolean values $p$, $q$, $r$, and $s$ on a 4-segment Boolean display.