Applications of Propositional Logic

Summary of Lecture

- Applications of Proposional Logic
  - ✓Translating English statements
  - ✓System Specifications
  - ✓Boolean Searches
  - ✓Logic Puzzles
  - ✓Logic Circuits
- Propositional Satisfiability

Translating English Sentences

- English Language is often ambiguous.
- Translation using Propositional Logic from English language to logical expressions
- Logical expressions can be analyzed to determine the truth values
- Rules of inference (discussed later in this unit) enables us to reason about the statements.

Some Examples of Translation

- “You can access the Internet from campus only if you are a computer science major or you are not a freshman.
- Equivalent logical expression would be

$$a \to (c \lor \neg f)$$

where a, c, and f represent “You can access the Internet from campus,” “You are a computer science major,” and “You are a freshman,” respectively

System Specifications

- Translating sentences in natural language(such as English) into logical expressions is an essential part of specifying both hardware and software systems.

- System and software engineers take requirements in natural language and produce precise and unambiguous specifications that can be used as the basis for system development.

System Specifications

- System specifications should be **consistent**
  - ✓ They should not contain conflicting requirements that could be used to derive a contradiction
- Example
  - Determine whether these system specifications are consistent:
    "The diagnostic message is stored in the buffer or it is retransmitted."
    "The diagnostic message is not stored in the buffer."
    "If the diagnostic message is stored in the buffer, then it is retransmitted."

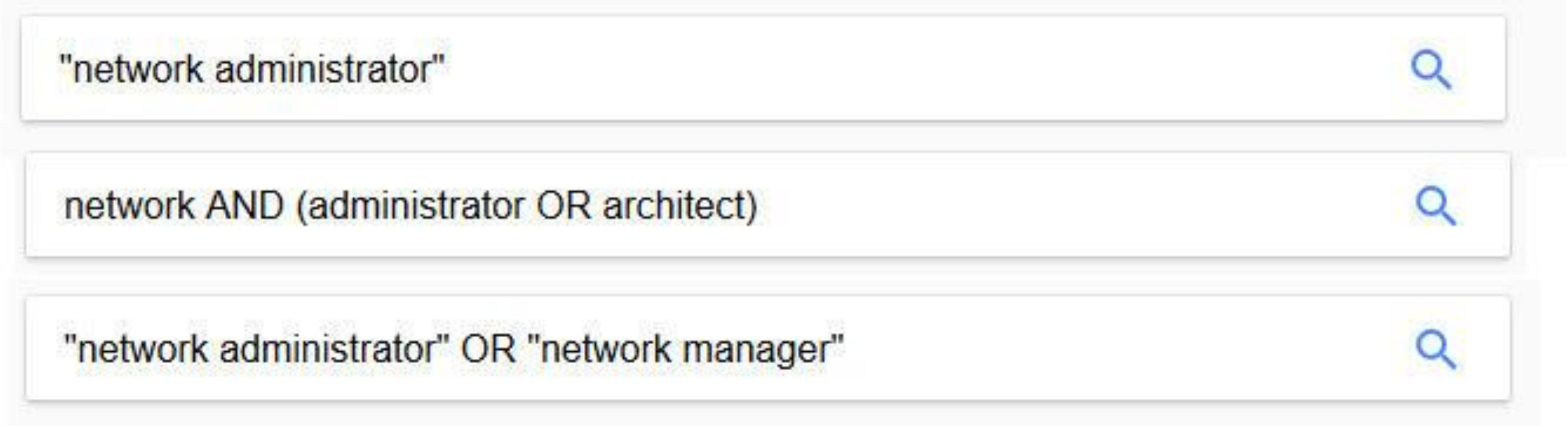
Solution …

- Let p denote “The diagnostic message is stored in the buffer” and let q denote “The diagnostic message is retransmitted.”
- The specification would be $p \vee q$, $\neg p$, and $p \rightarrow q$
- These specifications are consistent, because they are all true when p is false and q is true.

- Linked exercise
  - Do the system specifications in Example 4 remain consistent if the specification “The diagnostic message is not retransmitted” is added?

Boolean Searches

- Searching large collections of information, such as indexes of Web pages is computationally hard task.
- Boolean Searches
  - Use of connective AND, OR and NOT
- Example: Web page Searching



Logic Puzzles

- Logic puzzles can be solved using logical reasoning.
- Example

An island has two kinds of inhabitants, *knights*, who always tell the truth, and *knaves*, who always lie.

You go to the island and meet A and B.

- A says "B is a knight."
- B says "The two of us are of opposite types."

Example: What are the types of A and B?

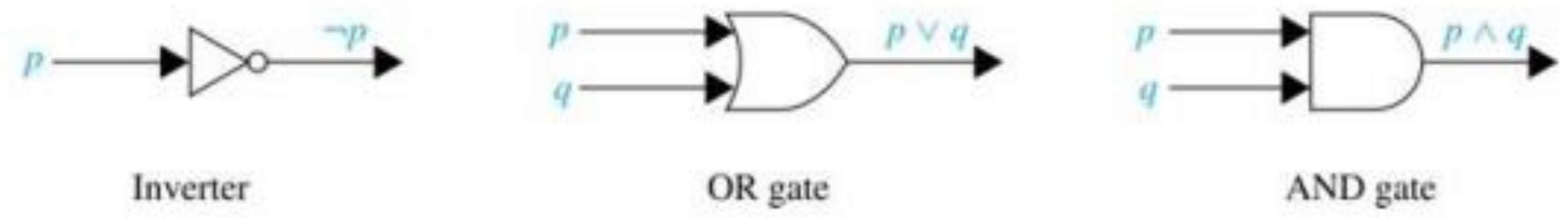
Logic Puzzles continued …

Solution: Let p and q be the statements that A is a knight and B is a knight, respectively. So, then $\neg p$ represents the proposition that A is a knave and $\neg q$ that B is a knave.

- If A is a knight, then p is true. Since knights tell the truth, q must also be true. Then $(p \land \neg q) \lor (\neg p \land q)$ would have to be true, but it is not. So, A is not a knight and therefore $\neg p$ must be true.
- If A is a knave, then B must not be a knight since knaves always lie. So, then both $\neg p$ and $\neg q$ hold since both are knaves.

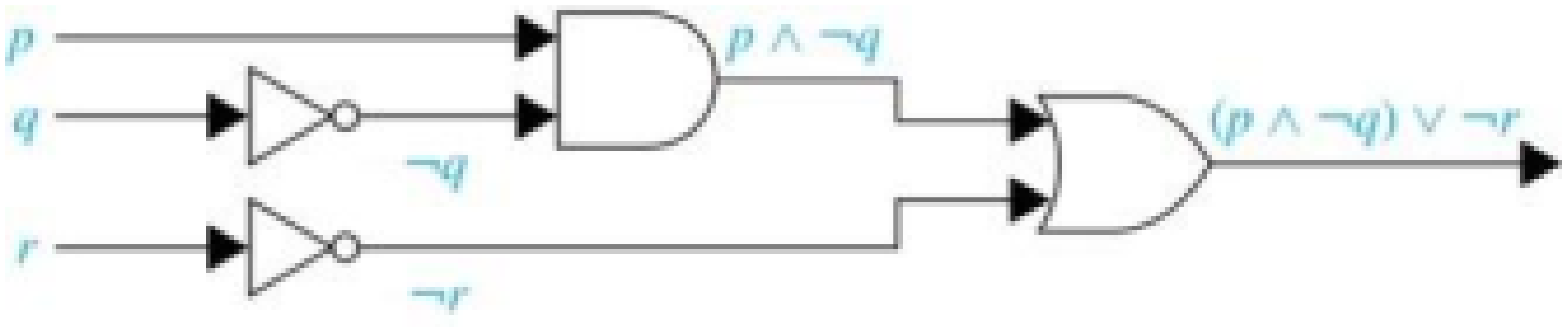
Logic Circuits

- Electronic circuits; each input/output signal can be viewed as a 0 or 1.
  - ✓0 represents **False**
  - ✓1 represents **True**
- Complicated circuits are constructed from three basic circuits called gates.



Logic Circuits … (will be discussed in detail in Unit 3)

- The inverter (**NOT gate**)takes an input bit and produces the negation of that bits.
- The **OR gate** takes two input bits and produces the value equivalent to the disjunction of the two bits.
- The **AND gate** takes two input bits and produces the value equivalent to the conjunction of the two bits
- More complicated digital circuits can be constructed by combining these basic circuits to produce the desired output given the input signals by building a circuit for each piece of the output expression and then combining them. For example:
-



Propositional Satisfiability

- A compound proposition is *satisfiable* if there is an assignment of truth values to its variables that make it true. When no such assignments exist, the compound proposition is *unsatisfiable*.

- A compound proposition is unsatisfiable if and only if its negation is a tautology.

Question on Propositional Satisfiability

- Determine the satisfiability of the following compound propositions:

$$(p \vee \neg q) \wedge (q \vee \neg r) \wedge (r \vee \neg p)$$

Question on Propositional Satisfiability

- Determine the satisfiability of the following compound propositions:

$$(p \vee \neg q) \wedge (q \vee \neg r) \wedge (r \vee \neg p)$$

- Solution: Satisfiable. Assign **T** to p, q and r.

Question on Propositional Satisfiability

- Determine the satisfiability of the following compound propositions:

$$\left(p \vee \neg q\right) \wedge \left(q \vee \neg r\right) \wedge \left(r \vee \neg p\right)$$

- Solution: Satisfiable. Assign **T** to p, q and r.
- Next Question: Determine the satisfiability of the following compound propositions:

$$\left(p \vee q \vee r\right) \wedge \left(\neg p \vee \neg q \vee \neg r\right)$$

Question on Propositional Satisfiability

- Determine the satisfiability of the following compound propositions:

$$(p \vee \neg q) \wedge (q \vee \neg r) \wedge (r \vee \neg p)$$

- Solution: Satisfiable. Assign **T** to p, q and r.
- Next Question: Determine the satisfiability of the following compound propositions:

$$(p \vee q \vee r) \wedge (\neg p \vee \neg q \vee \neg r)$$

Solution: Satisfiable. Assign **T** to *p* and ***F*** to *q*.

Question on Propositional Satisfiability …

- Determine the satisfiability of the following compound propositions:

$$(p \vee \neg q) \wedge (q \vee \neg r) \wedge (r \vee \neg p) \wedge (p \vee q \vee r) \wedge (\neg p \vee \neg q \vee \neg r)$$

Question on Propositional Satisfiability …

- Determine the satisfiability of the following compound propositions:

$$(p \vee \neg q) \wedge (q \vee \neg r) \wedge (r \vee \neg p) \wedge (p \vee q \vee r) \wedge (\neg p \vee \neg q \vee \neg r)$$

- **Solution:** Not satisfiable. Check each possible assignment of truth values to the propositional variables and none will make the proposition true.

Notations

$$\bigvee_{j=1}^{n} p_j \text{ is used for } p_1 \vee p_2 \vee \ldots \vee p_n$$

$$\bigwedge_{j=1}^{n} p_j \text{ is used for } p_1 \wedge p_2 \wedge \ldots \wedge p_n$$

Needed for the next example.

Sudoku

- A **Sudoku puzzle** is represented by a 9×9 grid made up of nine 3×3 subgrids, known as **blocks**. Some of the 81 cells of the puzzle are assigned one of the numbers 1,2, …, 9.
- The puzzle is solved by assigning numbers to each blank cell so that every row, column and block contains each of the nine possible numbers.
- Example

	2	9				4		
			5			1		
	4							
				4	2			
6							7	
5								
7			3					5
	1			9				
							6	

Encoding as a Satisfiability Problem

- Let $p(i,j,n)$ denote the proposition that is true when the number n is in the cell in the ith row and the jth column.
- There are $9 \times 9 \times 9 = 729$ such propositions.
- In the sample puzzle $p(5,1,6)$ is true, but $p(5,j,6)$ is false for $j = 2,3,\ldots 9$

Encoding as a Satisfiability Problem

For each cell with a given value, assert *p*(*i*,*j*,*n*), when the cell in row *i* and column *j* has the given value.

Assert that every row contains every number.

$$\bigwedge_{i=1}^{9}\bigwedge_{n=1}^{9}\bigvee_{j=1}^{9} p(i,j,n)$$

Assert that every column contains every number.

$$\bigwedge_{j=1}^{9}\bigwedge_{n=1}^{9}\bigvee_{i=1}^{9} p(i,j,n)$$

Encoding as a Satisfiability Problem

Assert that each of the 3 × 3 blocks contain every number.

$$\bigwedge_{r=0}^{2}\bigwedge_{s=0}^{2}\bigwedge_{n=1}^{9}\bigwedge_{i=1}^{3}\bigvee_{j=1}^{3} p\left(3r+i,3s+j,n\right)$$

(this is tricky - ideas from chapter 4 help)

Assert that no cell contains more than one number. Take the conjunction over all values of *n*, *n′*, *i*, and j, where each variable ranges from 1 to 9 and n ≠ *n′* , of

$$p\left(i,j,n\right)\rightarrow\neg p\left(i,j,n'\right)$$

- Home Exercise

  - Explain the steps in the construction of the compound proposition given in the text that asserts that every column of a 9×9 Sudoku puzzle contains every number.
  - Explain the steps in the construction of the compound proposition given in the text that asserts that each of the nine 3×3 blocks of a 9×9 Sudoku puzzle contains every number.

Solving Satisfiability Problems

- To solve a Sudoku puzzle, we need to find an assignment of truth values to the 729 variables of the form $p(i,j,n)$ that makes the conjunction of the assertions true. Those variables that are assigned T yield a solution to the puzzle.

- A truth table can always be used to determine the satisfiability of a compound proposition. But this is too complex even for modern computers for large problems.

- There has been much work on developing efficient methods for solving satisfiability problems as many practical problems can be translated into satisfiability problems.

Next lecture

- Predicates and Quantifiers …