Discrete Mathematics for Computer Science

Milos Hauskrecht
milos@cs.pitt.edu
5329 Sennott Square

Course administrivia

Instructor: Milos Hauskrecht
5329 Sennott Square
milos@cs.pitt.edu

TAs: Zitao Liu
5406 Sennott Square,
ztliu@cs.pitt.edu

Course web page:
http://www.cs.pitt.edu/~milos/courses/cs441/
Course administrivia

Lectures:
- Tuesdays, Thursdays: 11:00 AM - 12:15 PM
- 205 LAWRN

Recitations:
- held in 5313 SENSQ
  - Section 1: Thursdays 4:00 – 4:50 PM
  - Section 2: Fridays: 11:00 – 11:50 AM

Textbook:

Exercises from the book will be given for homework assignments.
Course administrivia

Grading policy

• Exams: (50%)
• Homework assignments: 40%
• Lectures/recitations: 10%

Weekly homework assignments

• Assigned in class and posted on the course web page
• Due one week later at the beginning of the lecture
• No extension policy

Collaboration policy:

• You may discuss the material covered in the course with your fellow students in order to understand it better
• However, homework assignments should be worked on and written up individually
Course administrivia

Course policies:
• Any un-intellectual behavior and cheating on exams, homework assignments, quizzes will be dealt with severely
• If you feel you may have violated the rules speak to us as soon as possible.
• Please make sure you read, understand and abide by the Academic Integrity Code for the Faculty and College of Arts and Sciences.

Course syllabus

Tentative topics:
• Logic and proofs
• Sets
• Functions
• Integers and modular arithmetic
• Sequences and summations
• Counting
• Probability
• Relations
• Graphs
Discrete mathematics

- **Discrete mathematics**
  - study of mathematical structures and objects that are fundamentally **discrete** rather than **continuous**.
- **Examples of objects** with discrete values are
  - **integers**, **graphs**, or **statements in logic**.
- Discrete mathematics and **computer science**.
  - Concepts from discrete mathematics are useful for describing **objects and problems in computer algorithms and programming languages**. These have applications in cryptography, automated theorem proving, and software development.
Course syllabus

Tentative topics:
• Logic and proofs
• Sets
• Functions
• Integers and modular arithmetic
• Sequences and summations
• Counting
• Probability
• Relations
• Graphs
Logic

Logic:
• defines a formal language for representing knowledge and for making logical inferences
• It helps us to understand how to construct a valid argument

Logic defines:
• Syntax of statements
• The meaning of statements
• The rules of logical inference (manipulation)

Propositional logic

• The simplest logic

• Definition:
  – A proposition is a statement that is either true or false.

• Examples:
  – Pitt is located in the Oakland section of Pittsburgh.
    • (T)
  – 5 + 2 = 8.
    • (F)
  – It is raining today.
    • (either T or F)
Propositional logic

- Examples (cont.):
  - How are you?
    - a question is not a proposition
  - \( x + 5 = 3 \)
    - since \( x \) is not specified, neither true nor false
  - 2 is a prime number.
    - (T)
  - She is very talented.
    - since she is not specified, neither true nor false
  - There are other life forms on other planets in the universe.
    - either T or F

Composite statements

- More complex propositional statements can be build from elementary statements using logical connectives.

Example:
- Proposition A: It rains outside
- Proposition B: We will see a movie
- A new (combined) proposition:
  If it rains outside then we will see a movie
Composite statements

- More complex propositional statements can be build from elementary statements using **logical connectives**.

- **Logical connectives:**
  - Negation
  - Conjunction
  - Disjunction
  - Exclusive or
  - Implication
  - Biconditional

Negation

**Definition:** Let \( p \) be a proposition. The statement "It is not the case that \( p \)." is another proposition, called the **negation of \( p \)**. The negation of \( p \) is denoted by \( \neg p \) and read as "not \( p \)."

**Example:**
- Pitt is located in the Oakland section of Pittsburgh.
  - \( \rightarrow \)
- It is **not the case** that Pitt is located in the Oakland section of Pittsburgh.

**Other examples:**
- \( 5 + 2 \neq 8 \).
- 10 is **not** a prime number.
- It is **not** the case that buses stop running at 9:00pm.
Negation

- **Negate the following propositions:**
  - It is raining today.
    - It is *not* raining today.
  - 2 is a prime number.
    - 2 is *not* a prime number
  - There are other life forms on other planets in the universe.
    - It is *not the case* that there are other life forms on other planets in the universe.

---

Negation

- A **truth table** displays the relationships between truth values (T or F) of different propositions.

<table>
<thead>
<tr>
<th>p</th>
<th>( \neg p )</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
</tr>
</tbody>
</table>

**Rows:** all possible values of elementary propositions:
Conjunction

• **Definition**: Let \( p \) and \( q \) be propositions. The proposition "\( p \text{ and } q \)" denoted by \( p \land q \), is true when both \( p \) and \( q \) are true and is false otherwise. The proposition \( p \land q \) is called the **conjunction** of \( p \) and \( q \).

• **Examples**:
  – Pitt is located in the Oakland section of Pittsburgh and \( 5 + 2 = 8 \)
  – It is raining today and \( 2 \) is a prime number.
  – \( 2 \) is a prime number and \( 5 + 2 \neq 8 \).
  – \( 13 \) is a perfect square and \( 9 \) is a prime.

Disjunction

• **Definition**: Let \( p \) and \( q \) be propositions. The proposition "\( p \text{ or } q \)" denoted by \( p \lor q \), is false when both \( p \) and \( q \) are false and is true otherwise. The proposition \( p \lor q \) is called the **disjunction** of \( p \) and \( q \).

• **Examples**:
  – Pitt is located in the Oakland section of Pittsburgh or \( 5 + 2 = 8 \).
  – It is raining today or \( 2 \) is a prime number.
  – \( 2 \) is a prime number or \( 5 + 2 \neq 8 \).
  – \( 13 \) is a perfect square or \( 9 \) is a prime.
Truth tables

- Conjunction and disjunction
- Four different combinations of values for $p$ and $q$

<table>
<thead>
<tr>
<th>$p$</th>
<th>$q$</th>
<th>$p \land q$</th>
<th>$p \lor q$</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
</tbody>
</table>

**Rows:** all possible combinations of values for elementary propositions: $2^n$ values

- NB: $p \lor q$ (the or is used inclusively, i.e., $p \lor q$ is true when either $p$ or $q$ or both are true).
Truth tables

- **Conjunction and disjunction**
- Four different combinations of values for p and q

<table>
<thead>
<tr>
<th>p</th>
<th>q</th>
<th>p \land q</th>
<th>p \lor q</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
</tbody>
</table>

- NB: p \lor q (the or is used inclusively, i.e., p \lor q is true when either p or q or both are true).

Exclusive or

- **Definition:** Let p and q be propositions. The proposition "p exclusive or q" denoted by p \oplus q, is true when exactly one of p and q is true and it is false otherwise.

<table>
<thead>
<tr>
<th>p</th>
<th>q</th>
<th>p \oplus q</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
</tbody>
</table>
Implication

• **Definition**: Let \( p \) and \( q \) be propositions. The proposition "\( p \) implies \( q \)" denoted by \( p \rightarrow q \) is called **implication**. It is false when \( p \) is true and \( q \) is false and is true otherwise.

• In \( p \rightarrow q \), \( p \) is called the **hypothesis** and \( q \) is called the **conclusion**.

\[
\begin{array}{ccc}
 p & q & p \rightarrow q \\
 T & T & T \\
 T & F & F \\
 F & T & T \\
 F & F & T \\
\end{array}
\]

• \( p \rightarrow q \) is read in a variety of equivalent ways:
  • if \( p \) then \( q \)
  • \( p \) only if \( q \)
  • \( p \) is sufficient for \( q \)
  • \( q \) whenever \( p \)

• **Examples**:
  – if Steelers win the Super Bowl in 2013 then 2 is a prime.
    • If F then T ?
Implication

• $p \rightarrow q$ is read in a variety of equivalent ways:
  • if $p$ then $q$
  • $p$ only if $q$
  • $p$ is sufficient for $q$
  • $q$ whenever $p$

• **Examples:**
  – if Steelers win the Super Bowl in 2013 then 2 is a prime.
    • $T$
  – if today is Tuesday then $2 * 3 = 8$.
    • What is the truth value?
Implication

- $p \rightarrow q$ is read in a variety of equivalent ways:
  - if $p$ then $q$
  - $p$ only if $q$
  - $p$ is sufficient for $q$
  - $q$ whenever $p$

- **Examples:**
  - if Steelers win the Super Bowl in 2013 then 2 is a prime.
    - T
  - if today is Tuesday then $2 \times 3 = 8$.
    - F

---

Implication

- The **converse** of $p \rightarrow q$ is $q \rightarrow p$.
- The **contrapositive** of $p \rightarrow q$ is $\neg q \rightarrow \neg p$
- The **inverse** of $p \rightarrow q$ is $\neg p \rightarrow \neg q$

- **Examples:**
  - If it snows, the traffic moves slowly.
    - $p$: it snows  $q$: traffic moves slowly.
    - $p \rightarrow q$
  - The **converse:**
    - If the traffic moves slowly then it snows.
      - $q \rightarrow p$
Implication

• The contrapositive of \( p \rightarrow q \) is \( \neg q \rightarrow \neg p \)

• The inverse of \( p \rightarrow q \) is \( \neg p \rightarrow \neg q \)

• Examples:
  • If it snows, the traffic moves slowly.
    – The contrapositive:
      • If the traffic does not move slowly then it does not snow.
        • \( \neg q \rightarrow \neg p \)
    – The inverse:
      • If it does not snow the traffic moves quickly.
        • \( \neg p \rightarrow \neg q \)

Biconditional

• Definition: Let \( p \) and \( q \) be propositions. The biconditional \( p \leftrightarrow q \) (read \( p \) if and only if \( q \)), is true when \( p \) and \( q \) have the same truth values and is false otherwise.

<table>
<thead>
<tr>
<th>( p )</th>
<th>( q )</th>
<th>( p \leftrightarrow q )</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
</tr>
</tbody>
</table>

• Note: two truth values always agree.
Constructing the truth table

- Example: Construct a truth table for 
  \((p \rightarrow q) \land (\neg p \leftrightarrow q)\)
- Simpler if we decompose the sentence to elementary and intermediate propositions

<table>
<thead>
<tr>
<th>p</th>
<th>q</th>
<th>\neg p</th>
<th>p \rightarrow q</th>
<th>\neg p \leftrightarrow q</th>
<th>(p \rightarrow q) \land (\neg p \leftrightarrow q)</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Rows: all possible combinations of values for elementary propositions: \(2^n\) values
Constructing the truth table

• Example: Construct the truth table for
  \((p \rightarrow q) \land (\neg p \leftrightarrow q)\)

<table>
<thead>
<tr>
<th>p</th>
<th>q</th>
<th>\neg p</th>
<th>p \rightarrow q</th>
<th>\neg p \leftrightarrow q</th>
<th>(p \rightarrow q) \land (\neg p \leftrightarrow q)</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
</tbody>
</table>

Typically the target (unknown) compound proposition and its values

Auxiliary compound propositions and their values

Constructing the truth table

• Examples: Construct a truth table for
  \((p \rightarrow q) \land (\neg p \leftrightarrow q)\)

<table>
<thead>
<tr>
<th>p</th>
<th>q</th>
<th>\neg p</th>
<th>p \rightarrow q</th>
<th>\neg p \leftrightarrow q</th>
<th>(p \rightarrow q) \land (\neg p \leftrightarrow q)</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
</tbody>
</table>
Constructing the truth table

- Examples: Construct a truth table for
  
  \((p \to q) \land (\neg p \leftrightarrow q)\)

\[
\begin{array}{|c|c|c|c|c|c|}
\hline
p & q & \neg p & p \to q & \neg p \leftrightarrow q & (p \to q) \land (\neg p \leftrightarrow q) \\
\hline
T & T & F & T & & \\
T & F & F & F & & \\
F & T & T & T & & \\
F & F & T & T & & \\
\hline
\end{array}
\]

Constructing the truth table

- Examples: Construct a truth table for
  
  \((p \to q) \land (\neg p \leftrightarrow q)\)

\[
\begin{array}{|c|c|c|c|c|c|}
\hline
p & q & \neg p & p \to q & \neg p \leftrightarrow q & (p \to q) \land (\neg p \leftrightarrow q) \\
\hline
T & T & F & T & F & \\
T & F & F & F & T & \\
F & T & T & T & T & \\
F & F & T & T & F & \\
\hline
\end{array}
\]
**Constructing the truth table**

- **Examples:** Construct a truth table for
  \[(p \rightarrow q) \land (\neg p \leftrightarrow q)\]

  Simpler if we decompose the sentence to elementary and intermediate propositions

<table>
<thead>
<tr>
<th>p</th>
<th>q</th>
<th>\neg p</th>
<th>p \rightarrow q</th>
<th>\neg p \leftrightarrow q</th>
<th>(p \rightarrow q) \land (\neg p \leftrightarrow q)</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
</tr>
</tbody>
</table>