Introduction

Discrete Structures

Goal: Understand how to use mathematics to reason about problems in Computer Sciences:
• sets and counting
• functions and relations
• sequences, summations
• logic
• proofs, including mathematical induction
• recurrences
• combinatorics
• trees
• graphs

What is “discrete” about Discrete Structures?
• Also called Discrete (or Finite) Mathematics
• Mathematics for Computer Science – finite in nature:
  – Algorithms
  – Data structures
  – Databases
  – Operating Systems
  – Computer Security
  – Digital imaging
• Unlike calculus which is concerned with infinite processes and the notion of continuity

Examples of problems solved using discrete structures
• How many ways are there to choose a valid password?
• How can we prove that a list of n numbers cannot be sorted using fewer than nlog₂n comparisons?
• Is there a link between two computer systems in a network?
• How many habitats do you need to create in a zoo so that animals don’t eat each other?
• What is a reasonable way to determine the significance of a webpage (page rank)
• Prove that in a gathering with 6 people, where each pair is either friends or strangers, there is a group of 3 people who are either all mutual friends or all mutual strangers.
Course organization

- Course website – links to all materials:
  http://www.csc.villanova.edu/~map/1300/s19/
- Lectures will cover approximately one chapter per week
- Exercises of various sorts:
  - Readiness or Warmup exercises
  - Drill exercises to review lecture
  - Homework problems
- 2 exams
- Final
- Piazza: questions, discussion, extra help

An old quote

A priest asked: What is Fate, Master?
And he answered:
It is that which gives a beast of burden its reason for existence.
It is that which men in former times had to bear upon their backs.
It is that which has caused nations to build byways from City
to City upon which carts and coaches pass, and alongside which
inns have come to be built to stave off Hunger, Thirst and
Weariness.
And that is Fate? said the priest.
Fate...I thought you said Freight, responded the Master.
That’s all right, said the priest. I wanted to know what Freight was too.

- Kehlog Albran

Source unknown: This quote appeared as one of the “fortunes” displayed by the fortune cookie program on old unix systems (“fortune”
was a program that ran automatically every time you logged out of the system and displayed a random, pithy saying.

Introduction to Logic

- Major themes
  - Sentences or propositions
  - Open sentences
  - Logical connectives
  - Truth tables
  - Converse and contrapositive
  - Logical equivalence, tautology, and contradiction

Why Logic?

Logic – a science of reasoning

- Basis of sound reasoning
  - gives precise meaning to mathematical (or other) statements
  - is used to distinguish between valid and invalid arguments

- Applications of logic in CS:
  - programming
  - design of hardware
  - verification of program correctness
  - modeling in artificial intelligence
  - inferences, integrity of databases
**Statement (or Proposition)**
A *declarative* sentence that is either **true** or **false**

**Are the following statements?**
- $1+2 = 3$
- today is my birthday
- New York is the capital of the USA
- $5 - 3 + 2$
- $x+y > 5$
- Is Villanova in Pennsylvania?
- Don’t talk
- Your feet are ugly
- This sentence is false

---

**Open Sentences**
- Declarative sentences containing *variable(s)* representing objects from some set $D$ called the *domain of discourse* (or just the *domain*)
- Truth of Open sentence depends on value(s) of variables
- Notation: $P(x)$ or $Q(x,y,z)$
- Also called *Propositional functions*

**Example**
Let $P(x)$ denote the statement “$x$ is even”.
**Domain of $x$: Positive Integers**

- $P(2)$
- $P(3)$

---

**Open Sentences**

**More Examples:**
- $x$ is enrolled in CSC 1051 and CSC 1300 this semester
- $x$ loves $y$
- $x$ loves $y$ and $y$ loves $z$
- $x$ is even or $x$ is odd

---

**Compound statements and Connectives**

*Compound statements* are formed from simpler statements using *connectives*, also called *logical operators*.

**The connectives we will study are:**
- *negation* or *not* operator denoted $\neg$ or $\sim$
- *conjunction* or *and* operator $\land$
- *disjunction* or *or* operator $\lor$
- *exclusive or* or *xor* operator $\oplus$
- *implication* $\rightarrow$
- *biconditional* $\leftrightarrow$
Negation (NOT)

If \( p \) is a statement, then “It is not the case that \( p \)” is also a statement, called the \textit{negation} of \( p \), denoted by \( \neg p \) (or \( \sim p \)) and read “not \( p \)” which is true when \( p \) is false, and is false when \( p \) is true.

**Example:** What is the negation of “Today is Wednesday”?

The truth table for negation:

<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>

Conjunction (AND)

The statement “\( p \) and \( q \)” denoted by \( p \land q \), is called the \textit{conjunction} of \( p \) and \( q \). It is true when both \( p \) and \( q \) are true, otherwise it is false.

**Examples:** Today is Wednesday and it is raining. Today is Wednesday but it is not raining.

The truth table for conjunction:

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

Disjunction (OR)

The statement “\( p \) or \( q \)” denoted by \( p \lor q \), is called the \textit{disjunction} of \( p \) and \( q \).
It is false when both \( p \) and \( q \) are false, otherwise it is true.

**Example:** Today is Sunday or a holiday.

The truth table for disjunction:

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

Exclusive OR (XOR)

The statement \( p \oplus q \) is called the \textit{exclusive or} of \( p \) and \( q \). It is true when exactly one of \( p \) and \( q \) is true, otherwise it is false.

**Example:** This dish comes with soup or salad.

The truth table for exclusive or:

<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>
Truth tables for more complex statements

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

Implication

The implication or conditional statement \(p \rightarrow q\) is the statement that is false only when \(p\) is true and \(q\) is false. \(p\) is called the hypothesis and \(q\) is called the conclusion.

The truth table for implication:

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

Readings for \(p \rightarrow q\):
- "if \(p\) then \(q\)"
- "\(p\) only if \(q\)"
- "\(q\) is necessary for \(p\)"
- "\(p\) is sufficient for \(q\)"
- "\(p\) implies \(q\)"
- "\(q\) if \(p\)"
- "\(q\) whenever \(p\)"

Examples of Implication Wording

If John is in L.A., then he is in California.
To be in California, it is sufficient for John to be in L.A.
To be in LA, it is necessary for John to be in California.

You will get an A if you study hard.
vs.
You will get an A only if you study hard.

More Examples of Implication wording:

If you place your order by 11:59pm December 21st, then we guarantee delivery by Christmas.
Placing your order by 11:59pm December 21st guarantees delivery by Christmas.
We guarantee delivery by Christmas if you place your order by 11:59pm December 21st.
More Examples of Implication wording:

If you place your order by 11:59pm December 21st, then we guarantee delivery by Christmas.

Placing your order by 11:59pm December 21st guarantees delivery by Christmas.

We guarantee delivery by Christmas if you place your order by 11:59pm December 21st.

Biconditional

The biconditional \( p \iff q \) is the statement that is true when \( p \) and \( q \) have the same truth values, and is false otherwise.

The truth table for biconditional:

\[
\begin{array}{c|c|c|c}
 p & q & p \iff q \\
 T & T & T \\
 T & F & F \\
 F & T & F \\
 F & F & T \\
\end{array}
\]

Readings for \( p \iff q \):

- “\( p \) if and only if \( q \)”
- “\( p \) is necessary and sufficient for \( q \)”
- “\( p \), then \( q \), and conversely”

Tautologies and contradictions

- \textit{tautology:} A statement that is always true (no matter what the truth values of the statements that occur in it)
  - The truth table for a tautology has “T” in every row.
- \textit{contradiction:} A statement that is always false
  - The truth table for a contradiction has “F” in every row.

Simplest example of a tautology and contradiction

\[
\begin{array}{c|c|c|c|c}
 p & \neg p & p \lor \neg p & p \land \neg p \\
 T & F & T & F \\
 F & T & T & F \\
\end{array}
\]

"The fish is fresh"
"The fish is fresh or the fish is not fresh"
"The fish is fresh and the fish is not fresh"
Tautology?

$p \land (p \lor q) \rightarrow q$

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

More examples of tautologies and contradictions?

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

Logical equivalence

We say that the statements $p$ and $q$ are logically equivalent (and write $p \equiv q$) if $p$ and $q$ have the same truth value for all combinations of truth values of their component statements.

• i.e., $p \equiv q$ just in case $p$ and $q$ have the same truth table

Examples:

$s \rightarrow u \equiv \neg s \lor u$, because:

<table>
<thead>
<tr>
<th>s</th>
<th>u</th>
<th>(s \rightarrow u)</th>
<th>(\neg s \lor u)</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>T</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
</tbody>
</table>

\(p \equiv \neg (\neg p)\), because:

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

De Morgan’s Laws for Logic

First De Morgan’s law for logic:

\(\neg (p \lor q) \equiv (\neg p) \land (\neg q)\)

Example: Negate: “Today is Sunday or a holiday”

Second De Morgan’s law for logic:

\(\neg (p \land q) \equiv\)

Example: Negate: “Today is Sunday and a holiday”
Biconditional, revisited:

Can you think of different, equivalent ways to express the biconditional $p \iff q$?

Some more examples:

- $p \rightarrow q \equiv \neg q \rightarrow \neg p$
- $(p \rightarrow q) \iff (\neg q \rightarrow \neg p)$ is a tautology
- $p \iff \neg p$ is a contradiction
- $(p \land (p \rightarrow q) \land (q \rightarrow r) \land (r \rightarrow s) \land (s \rightarrow w)) \rightarrow w$ is a tautology

Converse, Inverse and Contrapositive

- $q \rightarrow p$ is called the converse of $p \rightarrow q$
- $\neg p \rightarrow \neg q$ is called the inverse of $p \rightarrow q$
- $\neg q \rightarrow \neg p$ is called the contrapositive of $p \rightarrow q$

<table>
<thead>
<tr>
<th>$p$</th>
<th>$q$</th>
<th>$p \rightarrow q$</th>
<th>$q \rightarrow p$</th>
<th>$\neg p \rightarrow \neg q$</th>
<th>$\neg q \rightarrow \neg p$</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</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>F</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
</tbody>
</table>

Which of the above are logically equivalent?

Examples of Implication Wording (again):

Find the converse, inverse, & contrapositive

If John is in L.A., then he is in California.
To be in California, it is sufficient for John to be in L.A.
To be in LA, it is necessary for John to be in California.

You will get an A if you study hard.
vs.
You will get an A only if you study hard.