Course Staff

- **Instructor**: Prof. İlş Dillig
- **TA**: John Kallaugher
- **Proctor**: Jacob Van Geffen
- **Class meets every Tuesday, Thursday 5:00 pm - 6:15 pm**
- **Course webpage**: http://www.cs.utexas.edu/~isil/cs311h/
- **Contains contact info, office hours, slides from lectures, homework assignments etc.**

About this Course

- Give mathematical background you need for computer science
- **Topics**: Logic, proof techniques, number theory, combinatorics, graph theory, basic complexity theory . . .
- These will come up again and again in higher-level CS courses
  - Master CS311H material if you want to do well in future courses!

Textbook

- **Textbook (optional)**: Discrete Mathematics and Its Applications by Kenneth Rosen
- **Textbook not a substitute for lectures**: Class presentation may not follow book
- Skip many chapters and cover extra material

Piazza

- **Piazza page**: https://piazza.com/utexas/fall2016/cs311h/home
- **Homework #0**: Make sure you can access Piazza page!
- Please post class-related questions on Piazza instead of emailing instructor TA’s
  - If something is not clear to you, it won’t be to others either
  - You’ll get answers a lot quicker
- Please use common sense when posting questions on Piazza
  - Hints/ideas ok, but cannot post full solutions!!

Discussion Sections and Office Hours

- **Discussion sections on Mondays and Wednesdays 2-3 pm**
- Discussion section will answer questions, solve new problems, and go over previous homework
- Isil’s office hours: Tuesday/Thursdays 6:15-7 pm (GDC 5.726)
- John’s office hours: Mon 3-5pm (location TBD)
- Jacob’s office hours: Wed 3-5pm (location TBD)
Requirements

▶ Weekly written homework assignments
▶ Two midterm exams: in-class, closed-book
  ▶ Allowed to bring 3 pages of hand-prepared notes
▶ Scheduled for September 29, November 10
▶ Final exam on December 8 (7-10 pm)
▶ No make-up exams given unless you have serious, documented medical emergency

Grading

▶ Final exam: 40% of final grade
▶ Each midterm: 20% of final grade
▶ Homework: 20% of final grade
▶ Final grades may be curved, but lower bounds guaranteed (e.g., get at least A- if grade is 90% or higher)

Homework Policy

▶ Homework due at the beginning of class on due date
  ▶ No credit unless turned in by 5 PM on due date
  ▶ Late submissions not allowed, but lowest homework score dropped when calculating grades
▶ We will only give hard copies of homework solutions in class

Honor Code

▶ Homework write-up must be your own
▶ May not copy answers from on-line resources or other students
▶ If you discuss with others, write-up must mention their names
▶ Honor code taken very seriously at UT
  ▶ May be expelled for violating honor code!
  ▶ Please read departmental guidelines (link from course webpage)

Class Participation

▶ Everyone expected to attend lectures
  ▶ Please do not use cell phone, laptop, or tablets during lecture!
▶ Ask questions!
  ▶ No question is a stupid question!
  ▶ Other students also benefit from your questions
▶ Make class fun by participating!
  ▶ Might win chocolate if you answer questions :)
Logic
▶ Logic: study of valid reasoning; fundamental to CS
▶ Allows us to represent knowledge in precise, mathematical way
▶ Allows us to make valid inferences using a set of precise rules
▶ Many applications in CS:
  AI, programming languages, databases, computer architecture, automated testing and program analysis, . . .

Propositional Logic
▶ Simplest logic is propositional logic
▶ Building blocks of propositional logic are propositions
▶ A proposition is a statement that is either true or false
▶ Examples:
  - "CS311 is a course in discrete mathematics": True
  - "Austin is located in California": False
  - "Pay attention": Not a proposition
  - "x+1 =2": Not a proposition

Propositional Variables, Truth Value
▶ Truth value of a proposition identifies whether a proposition is true (written T) or false (written F)
▶ What is truth value of "Today is Friday"? F
▶ Variables that represent propositions are called propositional variables
▶ Denote propositional variables using lower-case letters, such as p, p₁, p₂, q, r, s, . . .
▶ Truth value of a propositional variable is either T or F.

Conjunction
▶ Conjunction of two propositions p and q, written p \(\land\) q, is the proposition "p and q"
▶ p \(\land\) q is T if both p is true and q is true, and F otherwise.
▶ What is the conjunction and the truth value of p \(\land\) q for . . .
  - p = "It is fall semester", q = "Today is Thursday"?
  - p = "It is Thursday", q = "It is morning"?
Disjunction

- Disjunction of two propositions \( p \) and \( q \), written \( p \lor q \), is the proposition "\( p \) or \( q \)"
- \( p \lor q \) is \( T \) if either \( p \) is true or \( q \) is true, and \( F \) otherwise.
- What is the disjunction and the truth value of \( p \lor q \) for . . .
  - \( p = \) "It is spring semester", \( q = \) "Today is Thursday"
  - \( p = \) "It is Friday", \( q = \) "It is morning"

Propositional Formulas and Truth Tables

- Truth table for propositional formula \( F \) shows truth value of \( F \) for every possible value of its constituent atomic propositions
- Example: Truth table for \( \neg p \)
<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>

Example: Truth table for \( p \lor q \)

<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>

Constructing Truth Tables

Useful strategy for constructing truth tables for a formula \( F \):
1. Identify \( F \)'s constituent atomic propositions
2. Identify \( F \)'s compound propositions in increasing order of complexity, including \( F \) itself
3. Construct a table enumerating all combinations of truth values for atomic propositions
4. Fill in values of compound propositions for each row

Examples

Construct truth tables for the following formulas:
1. \( (p \lor q) \land \neg p \)
2. \( (p \land q) \lor (\neg p \land \neg q) \)
3. \( (p \lor q \lor \neg r) \land r \)

More Logical Connectives

- \( \land, \lor, \neg \) most common boolean connectives, but there are other boolean connectives as well
- Other connectives: exclusive or \( \oplus \), implication \( \to \), biconditional \( \leftrightarrow \)
- Exclusive or: \( p \oplus q \) is true when exactly one of \( p \) and \( q \) is true, and 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>

Truth table:

Implication (Conditional)

- An implication (or conditional) \( p \to q \) is read "if \( p \) then \( q \)" or "\( p \) implies \( q \)"
- It is false if \( p \) is true and \( q \) is false, and true otherwise
- Exercise: Draw truth table for \( p \to q \)
- In an implication \( p \to q \), \( p \) is called antecedent and \( q \) is called consequent
Converting English into Logic

Let \( p = " I major in CS" \) and \( q = " I will find a good job" \). How do we translate following English sentences into logical formulas?

- "If I major in CS, then I will find a good job": \( p \rightarrow q \)
- "I will not find a good job unless I major in CS": \( \neg q \rightarrow \neg p \)
- "It is sufficient for me to major in CS to find a good job": \( p \rightarrow q \)
- "It is necessary for me to major in CS to find a good job": \( q \rightarrow p \)

More English - Logic Conversions

Let \( p = " I major in CS" \), \( q = " I will find a good job" \), and \( r = " I can program" \). How do we translate following English sentences into logical formulas?

- " I will not find a good job unless I major in CS or I can program": \( \neg q \rightarrow (p \lor r) \)
- " I will not find a good job unless I major in CS and I can program": \( \neg q \rightarrow (p \land r) \)
- "A prerequisite for finding a good job is that I can program": \( p \rightarrow r \)
- "If I major in CS, then I will be able to program and I can find a good job": \( p \rightarrow (r \land q) \)

Converse of a Implication

- The converse of an implication \( p \rightarrow q \) is \( q \rightarrow p \).
- What is the converse of "If I am a CS major, then I can program"?
- What is the converse of "If I get an A in CS311, then I am smart"?
- Note: It is possible for a implication to be true, but its converse to be false, e.g., \( F \rightarrow T \) is true, but converse false.

Inverse of an Implication

- The inverse of an implication \( p \rightarrow q \) is \( \neg p \rightarrow \neg q \).
- What is the inverse of "If I am a CS major, then I can program"?
- What is the inverse of "If I get an A in CS311, then I am smart"?
- Note: It is possible for a implication to be true, but its inverse to be false. \( F \rightarrow T \) is true, but inverse is false.

Contrapositive of Implication

- The contrapositive of an implication \( p \rightarrow q \) is \( \neg q \rightarrow \neg p \).
- What is the contrapositive of "If I am a CS major, then I can program"?
- What is the contrapositive of "If I get an A in CS311, then I am smart"?
- Question: Is it possible for an implication to be true, but its contrapositive to be false?

Conditional and its Contrapositive

A conditional \( p \rightarrow q \) and its contrapositive \( \neg q \rightarrow \neg p \) always have the same truth value.

- Prove it!
Question

- Given $p \rightarrow q$, is it possible that its converse is true, but inverse is false?

Summary

- Conditional is of the form $p \rightarrow q$
- Converse: $q \rightarrow p$
- Inverse: $\neg p \rightarrow \neg q$
- Contrapositive: $\neg q \rightarrow \neg p$
- Conditional and contrapositive have same truth value
- Inverse and converse always have same truth value

Biconditionals

- A biconditional $p \leftrightarrow q$ is the proposition "$p$ if and only if $q$".
- The biconditional $p \leftrightarrow q$ is true if $p$ and $q$ have same truth value, and false otherwise.
- Exercise: Construct a truth table for $p \leftrightarrow q$
- Question: How can we express $p \leftrightarrow q$ using the other boolean connectives?

Operator Precedence

- Negation ($\neg$) has higher precedence than all other connectives.
- Question: Does $\neg p \land q$ mean (i) $\neg(p \land q)$ or (ii) $(\neg p) \land q$?
- Conjunction ($\land$) has next highest precedence.
- Question: Does $p \land q \lor q$ mean (i) $(p \land q) \lor r$ or (ii) $p \land (q \lor r)$?
- Disjunction ($\lor$) has third highest precedence.
- Next highest is precedence is $\rightarrow$, and lowest precedence is $\leftrightarrow$

Operator Precedence Example

- Which is the correct interpretation of the formula

$$p \lor q \land r \leftrightarrow q \rightarrow \neg r$$

(A) $(p \lor (q \land r)) \leftrightarrow q \rightarrow (\neg r)$
(B) $(p \lor q) \land r \leftrightarrow q \rightarrow (\neg r)$
(C) $(p \lor (q \land r)) \leftrightarrow (q \rightarrow (\neg r))$
(D) $(p \lor ((q \land r) \leftrightarrow q)) \rightarrow (\neg r)$
Summary

- Formulas in propositional logic are formed using propositional variables and boolean connectives
- **Connectives**: negation $\neg$, conjunction $\land$, disjunction $\lor$, conditional $\rightarrow$, biconditional $\leftrightarrow$
- Truth table shows truth value of formula under all possible assignments to variables