

CS311H: Discrete Mathematics

Intro and Propositional Logic

Instructor: Işıl Dillig

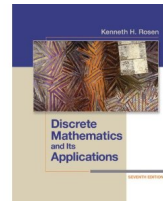
Course Staff

- ▶ **Instructor:** Prof. Işıl Dillig
- ▶ **Proctors:** Maruth Goyal, Abby Criswell, Rahul Krishnan, Anand Iyer
- ▶ **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/fall2020/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!!
- ▶ If you have a more personal question, ok to use email (cs311h-staff@cs.utexas.edu)

Discussion Sections and Office Hours

- ▶ Discussion sections on Fridays 1-2 pm and 2-3 pm
- ▶ Discussion section will answer questions, solve new problems, and go over previous homework
- ▶ Işıl's OH: Thursdays 5-6 pm (Zoom link on Canvas)
- ▶ Maruth's OH: Tuesday 3:30-4:30 pm
- ▶ Abby's OH: Friday 12-1pm
- ▶ Rahul's OH: Wednesday 2-3 pm
- ▶ Anand's OH: Monday 3:30-4:30 pm

Requirements

- ▶ Exams + problem sets + class attendance/participation
- ▶ Exams are take-home and cumulative (three total)
 - ▶ Released on Tuesdays and due on Thursdays before class
 - ▶ Scheduled to be released on Sep 29, Oct 27, and Dec 1
- ▶ 10 problem sets (about once every week)

Grading

- ▶ **Exam:** collectively 50% of final grade
- ▶ **Homework:** 45% of final grade
- ▶ **Attendance/participation:** 5% of final grade
- ▶ Final grades will be curved

Homework Policy

- ▶ Homework must be submitted by **1:30 pm** on the due date
- ▶ Late submissions **not** allowed, lowest homework score dropped when calculating grades
- ▶ Homework must be done **on your own**, but allowed to ask questions on Piazza and during office hours
 - ▶ **Not allowed** to do problem sets in groups
 - ▶ **Not allowed** to check solutions with each other
- ▶ Homework solutions must be typeset using Latex and submitted through Gradescope
- ▶ Two of the homework assignments will involve programming, rest are problem sets

More on Homework

- ▶ Problem sets in this class will be much harder than what you are used to from high school!!
 - ▶ Normal to spend >30mins on a single HW question
 - ▶ Do not seek help from us unless you've spent at least one hour on each problem
- ▶ Expect each problem set to take > 6 hours

Exam Policy

- ▶ All exams are take-home exams and will be cumulative
- ▶ Can only consult lecture notes and textbook but nothing else
- ▶ Cannot talk with others about exam questions (no discussion allowed, including on Piazza!)
- ▶ If you have clarification question about exam, please email cs311h-staff.cs.utexas.edu
- ▶ Honor code taken very seriously at UT (both for exams + ps)
 - ▶ May be expelled for violating honor code!
 - ▶ Please read departmental guidelines (link from course webpage)

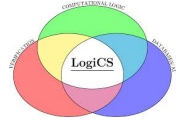
Class Participation

- ▶ Everyone expected to attend lectures and **turn video on**
- ▶ 5% of course grade for participation (attendance, asking/answering questions)
- ▶ Please ask questions!
 - ▶ Will make class more fun for everyone
 - ▶ Others also benefit from your questions

Let's get started!

Logic

- ▶ Logic: study of valid reasoning; fundamental to CS
- ▶ Allows us to represent knowledge in a formal/mathematical way and automate some types of reasoning
- ▶ **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"?
- ▶ Variables that represent propositions are called **propositional variables**
- ▶ Denote propositional variables using lower-case letters, such as $p, p_1, p_2, q, r, s, \dots$
- ▶ Truth value of a propositional variable is either T or F.

Compound Propositions

- ▶ More complex propositions formed using **logical connectives** (also called **boolean connectives**)
- ▶ Three basic logical connectives:
 1. \wedge : **conjunction** (read "and"),
 2. \vee : **disjunction** (read "or")
 3. \neg : **negation** (read "not")
- ▶ Propositions formed using these logical connectives called **compound propositions**; otherwise **atomic propositions**
- ▶ A **propositional formula** is either an atomic or compound proposition

Negation

- ▶ Negation of a proposition p , written $\neg p$, represents the statement "It is not the case that p ".
- ▶ If p is **T**, $\neg p$ is **F** and vice versa.
- ▶ In simple English, what is $\neg p$ if p stands for ...
 - ▶ "Less than 80 students are enrolled in CS311"?

Conjunction

- ▶ **Conjunction** of two propositions p and q , written $p \wedge q$, is the proposition "p and q"
- ▶ $p \wedge q$ is T if **both** p is true **and** q is true, and F otherwise.
- ▶ What is the conjunction and the truth value of $p \wedge q$ for ...
 - ▶ $p =$ "It is Thursday", $q =$ "It is morning" ?

Disjunction

- ▶ **Disjunction** of two propositions p and q , written $p \vee q$, is the proposition "p or q"
- ▶ $p \vee q$ is T if **either** p is true **or** q is true, and F otherwise.
- ▶ What is the disjunction and the truth value of $p \vee q$ for ...
 - ▶ $p =$ "It is spring semester", $q =$ "Today is Thursday"?

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 $\neg p$

p	$\neg p$
T	F
F	T

- ▶ **Example:** Truth table for $p \vee q$

p	q	$p \vee q$
T	T	T
T	F	T
F	T	T
F	F	F

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. $(p \vee q) \wedge \neg p$
2. $(p \wedge q) \vee (\neg p \wedge \neg q)$
3. $(p \vee q \vee \neg r) \wedge r$

More Logical Connectives

- ▶ \wedge, \vee, \neg most common boolean connectives, but there are other boolean connectives as well
- ▶ Other connectives: **exclusive or** \oplus , **implication** \rightarrow , **biconditional** \leftrightarrow
- ▶ **Exclusive or:** $p \oplus q$ is true when exactly one of p and q is true, and false otherwise

- ▶ Truth table:

p	q	$p \oplus q$
T	T	F
T	F	T
F	T	T
F	F	F

Implication (Conditional)

- ▶ An **implication** (or conditional) $p \rightarrow q$ 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 $p \rightarrow q$
- ▶ In an implication $p \rightarrow q$, 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":
- ▶ "I will not find a good job unless I major in CS":
- ▶ "It is sufficient for me to major in CS to find a good job":
- ▶ "It is necessary for me to major in CS to find a good job":

More English - Logic Conversions

Let $p =$ "I major in CS", $q =$ "I will find a good job", $r =$ "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":
- ▶ "I will not find a good job unless I major in CS and I can program":
- ▶ "A prerequisite for finding a good job is that I can program":
- ▶ "If I major in CS, then I will be able to program and I can find a good job":

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"?
- ▶ **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 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"?
- ▶ **Question:** Is it possible for an implication to be true, but its contrapositive to be false?

Question

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

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

- ▶ Given a formula $p \wedge q \vee r$, do we parse this as $(p \wedge q) \vee r$ or $p \wedge (q \vee r)$?
- ▶ Without settling on a convention, formulas without explicit paranthesization are ambiguous.
- ▶ To avoid ambiguity, we will specify **precedence** for logical connectives.

Operator Precedence, cont.

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

Operator Precedence Example

- ▶ Which is the correct interpretation of the formula

$$p \vee q \wedge r \leftrightarrow q \rightarrow \neg r$$

- (A) $((p \vee (q \wedge r)) \leftrightarrow q) \rightarrow (\neg r)$
- (B) $((p \vee q) \wedge r) \leftrightarrow q \rightarrow (\neg r)$
- (C) $(p \vee (q \wedge r)) \leftrightarrow (q \rightarrow (\neg r))$
- (D) $(p \vee ((q \wedge r) \leftrightarrow q)) \rightarrow (\neg r)$