What is a Proof?

1 Course Overview

1.1 Course Goals and Content

This course is actually a combination of three "miniature" courses: thinking (9 lectures), counting (8), and gambling (9). We aim to cover:

Basic discrete math concepts. Basic data types and structures used in programming and in reasoning about computer science (numbers, Booleans, strings; sets, functions; relations, trees, graphs,...). Their properties.

Proofs: How to do them, how to read them, how to recognize correct/incorrect ones.

Reasoning about propositions, predicates (with quantifiers).

Basic proof strategies: Direct proof, contradiction, cases... Induction (very important in this course and other places in CS).

We'll be working on this throughout the term.

Mathematical foundations for software engineering (6170). In 6.170, one major element is formal specification and verification of software systems.

We'll cover the underlying concepts and methods.

Develop models for programs and their required behavior.

Prove that a program satisfies its requirements.

Analytic Tools. We'll cover methods for handling series, sums, recurrences, asymptotic analysis, counting, and other problems in discrete math.

Much more than calculus, this is the kind of math that tends to arise in computer science problems.

A lot of this gets used in 6.046

Discrete probability (6046, 6033) . We'll show how to analyze random events.

Critical in the study of real-world computer systems.

Probability provides good ways to model the unpredictability of the real world.

The course also supports 6045 (theory of computation).

Overall, we want you to learn how to use mathematics to help you think about concepts in computer science.

- define abstract models for real-world objects like programs, computers, communication channels.
- Reason about these models carefully, correctly.
- Use the conclusions to help understand the real-world objects.)

Careful proofs are associated with virtually everything covered in the course: proofs in class, proofs in homework, lots of examples of pitfalls in writing proofs ...

The course moves rapidly from topic to topic.

- Disadvantage: none last long enough for thorough coverage.
- Advantage: boredom/confusion is temporary.

2 What are proofs? Why do we do them?

What is a proof?

Layman's definition of proof: "A method of ascertaining the truth."

There are many ways to decide what's true in society and even in science:

Law: judge and jury. Truth is by majority opinion.

Religion: Truth is by inner conviction.

Science: evidence of experiment/observation (e.g., General Relativity, The Big Bang).

Statistics: sampling (testing examples).

Reputable authority: truth accepted because someone says so (e.g., the professor). They often base their claim on one of the above methods

Boss: Truth accepted from someone you shouldn't disagree with.

"I don't see why not." Shift the burden of proof to someone who disagrees with you.

But these methods can go wrong:

- Conviction: There aren't any bugs in *my* programs. . . .
- Authority: Intel says the Pentium is fine. Stalin said evolution was wrong.
- Religion: whose?

General problem: these methods don't convince everyone.

So mathematics focuses on a particularly convincing kind of argument: a *proof.*

Definition 2.1 A *proof* is a verification of a *proposition* by a chain of *logical deductions* from a base set of *axioms*.

The pieces of this definition are:

Definitions and syntax: Agreement on common language of discussion

Propositions. Precise statement of what your are proving

Axioms. Careful listing of what you are assuming in your proof

Rules of deduction. Agreement on ways to derive new truths from old ones.

In the rest of this lecture, I'll explain what these things are, present the most basic proof methods, and give examples. The basic idea is to specify everything so precisely that everyone is convinced. The main benefit: if there is agreement on assumptions and deduction rules, that forces agreement on conclusions as well. Also, even if we can't understand the entire argument as once, if we can follow/believe each deductive step, we can be comfortable accepting conclusions

By learning to read and do proofs, you will be able to

- Convince others
- Find flaws in others' attempts to convince you
- Find flaws in your own proofs, so can figure out what else you need to find out or assume

3 Basic data types

Reading: Velleman chapter 1, Rosen 1.1-1.5

As a first step, we need to agree on some basic language of discourse so we can say things to each other. These are our "nouns." We'll be using all the standard basic data types used in programming: numbers, Booleans (true/false), and strings. We'll also use other basic data types from mathematics, such sets and functions. These are used in reasoning about programs. This is review for most of you, so I won't lecture about it. I'll just list the basic notions: definitions, operations, some properties. You will be expected to read this material carefully on your own, making sure you can do all the study exercises.

3.1 Booleans

Reading: Velleman 1.1–1.2, Rosen 1.1

Values: 1 (true), 0 (false)
Operations: and ($\wedge$), or ($\vee$), not ($\neg$), exclusive-or ($\oplus$).

These operations can be explained with tables:

x	$\neg x$
1	0
0	1

x	y	$x \wedge y$	$x \vee y$	$x \oplus y$
1	1	1	1	0
1	0	0	1	1
0	1	0	1	1
0	0	0	0	0

3.2 Numbers

Reading: Rosen, Section 1.4

These are just the standard number classes.

The Natural numbers $N = \{0, 1, 2, \dots\}$

Most math books actually let N start at 1, not 0, but computer scientists like to start counting with 0. We'll stay consistent with the texts. I'll write N^+ for the positive naturals.

Operations: $+$, $-$ (partial), times, integer division (round down), remainder (mod), exponentiation

The Integers, denoted by Z: N and its negatives

The Rationals, denoted by Q

These are all numbers expressible as a ratio of two integers $\frac{a}{b}$, where $b \neq 0$. A basic theorem (we might prove it later) says that we can write any rational number in lowest terms.

The Reals, denoted by R

3.3 Strings

Strings are sequences of symbols drawn from some fixed alphabet. For example, if the alphabet consists of a, b, c,some strings are abc, aaa, b, andthe empty string (λ, no symbols). The main interesting operation is concatenation. Concatenating the two strings abc and bba gives the string $abcbba$.

3.4 Sets

Reading: Velleman 1.3–1.4, Rosen 1.4–1.5

Terms are $\emptyset$, universal set, subset, proper subset. Operations are: union ($\cup$), intersection ($\cap$), complement (overbar). Venn diagrams are useful for thinking about sets. There are many laws about how the operations behave—see Rosen p. 49. E.g.: $A \cup A = A$.

3.5 Functions

Reading: Rosen, Section 1.6

A functions is a rule that assigns an element of one set (range, codomain) to each element of another set (domain).
Properties of functions:
One-to-one (no two domain elements get the same range element)

Onto (every element of the range gets used)

Operations include function inverse and composition ($f \circ g(x) = f(g(x))$)

4 Propositions

Now we have a language. Let's use it to say some things we might want to convince people about.

4.1 What are propositions?

Definition 4.1 A *proposition* is a statement that has a truth value, *true* or *false*.

Examples:

1. $2 + 3 = 5$ is a prime number.

   This is a proposition, and it's true.

2. 8 is a prime number.

   This is a false proposition.

3. x is a prime number.

   This is not a proposition. It has no truth value, because we don't know what x is. It's said to be an *unbound variable*.

4. There is an least one prime number.

   This is a true proposition. In math notation, it can be written as $\exists x(x \text{ is a prime})$. The quantifier $\exists$ "binds" x.

5. All natural numbers ≥ 2 are primes.

   This is a false proposition. It can be written as $\forall x \in N, x \geq 2(\ldots)$. The quantifier $\forall$ also binds x.

   We will revisit the quantifiers soon (they have their own reasoning methods). For now, we are just using them as shorthand notation for the English.

6. All x of the form $n^2 + n + 41$, where $n \in N$, are prime.

   This is a proposition, but it is not so immediate to determine if true or false. Certainly the preceding proposition is true for many natural numbers n:

$$
\begin{array}{cc}
n = 0 \rightarrow 41 & (prime) \\
n = 1 \rightarrow 43 & (prime) \\
n = 2 \rightarrow 47 & (prime) \\
n = 3 \rightarrow 53 & (prime) \\
\ldots & (more\ primes) \\
n = 20 \rightarrow 461 & (prime) \\
\ldots & (still\ more\ primes) \\
n = 39 \rightarrow 1601 & (prime)
\end{array}
$$

But if $n = 40$, then $n^2 + n + 41 = 40^2 + 40 + 41 = 41 \cdot 41$, which is not prime. Since the expression is not prime *for all* n, the proposition is false!

7. The terms "true" and "false" are regarded as special cases of propositions, with the obvious truth values.

8. $\forall A, B(\overline{A} \cup \overline{B} = \overline{A \cup B})$.

   Propositions can be about other things besides numbers, e.g., sets. This one is false.

9. $\forall A, B(\overline{A} \cup \overline{B} = \overline{A \cap B})$.
   $\forall A, B(\overline{A} \cap \overline{B} = \overline{A \cup B})$.

   These are called De Morgan's laws and are true.

10. The original Pentium chip always divided correctly.

    Intel's "proofs" by authority and by sampling turned out to be invalid. The proposition is false.

11. White can always win at chess.

    A proposition whose truth value nobody knows.

12. Every even integer greater than 2 is the sum of two prime numbers.

    No one knows whether this proposition is true or false. This is the Goldbach Conjecture, which dates back to 1742. It was raised in *The Boston Globe* as "one of the great unsolved mysteries."Unfortunately their "example" was $20 = 9 + 11$!

13. There is no $x, y, z \in N^+, n \geq 3$ such that $x^n + y^n = z^n$.

    This is Fermat's last theorem, 1640s. He claimed but never proved it. Only recently have we learned for sure that it's true. Andrew Wiles, in 1994, gave a 130-page proof. After months of work, other mathematicians found bugs. Wiles needed 100s more pages to fix them. But even while we didn't know if it was true or false, it was still a proposition.

14. $a^4 + b^4 + c^4 = d^4$ has no positive integer solutions.

    This statement was conjectured by Euler (1769). The first counterexample was found by Elkies, 218 years later. The smallest one: $95800^4 + 217519^4 + 414560^4 = 422481^4$.

15. Every map can be colored with four colors, so that no two adjacent regions have the same color (Four Color Theorem).

    This proposition was first "proved" in 1879. The "proof" stood for 10 years before a flaw was found. Recently, the theorem proved with computer aid. People still debate whether this is a valid proof. After all, nobody has prove that the computer is working correctly!

Not all sentences are propositions:

- $+x * *z = f$.
- What time is it?
- This sentence is false.

Propositions involve previously *defined* terms:

- prime, map, triangle, chess, fair coin, probability.
- $+, =, 1, \geq$.

4.2 Compound propositions and truth tables

Logical connectives are used to combine simple propositions. I already discussed and ($\wedge$), or ($\vee$), exclusive-or ($\oplus$), not ($\neg$) for Booleans. There is also implication (if-then, $\Longrightarrow$), double implication (equivalence, $\Leftrightarrow$). These are very important in logical reasoning.

The truth values of compound propositions are obtained from the truth values of the simpler propositions according to the *truth tables*:

P	$\neg P$
T	F
F	T

If a proposition P is true, its negation is false, and vice versa.

P	Q	$P \wedge Q$	$P \vee Q$	$P \oplus Q$
T	T	T	T	F
T	F	F	T	T
F	T	F	T	T
F	F	F	F	F

Import: $\vee$ is **inclusive** or. It is true if either *or both* of the two statements are true. Sometimes in English or is exclusive: Karger or Lynch will teach today is saying one but not both.

P	Q	$P \Rightarrow Q$	$P \Leftrightarrow Q$
T	T	T	T
T	F	F	F
F	T	T	F
F	F	T	T

The nice thing about these truth tables is that it doesn't matter what the specific propositions are; we know how to get compound truth value from basic truth values.

The only slightly mysterious column here is the implication column. We say that $P \Rightarrow Q$ is false only if the hypothesis P is true and the conclusion Q is false. Some thought shows this aligns with our intuition. Consider the statement "if you do your homework, then I will give you an A". Under what circumstances would you accuse me of lying? Well, I am clearly lying if you do your homework but then you don't get an A. But if you skip your homework and I give you an A *anyway,* you won't say I'm lying. More generally, if you skip your homework the statement promises nothing, so it is true regardless of what I do.

One way to determine truth values for compound propositions is by adding new columns to the basic truth table(s). Consider for example

$$R = (P \Rightarrow Q) \Leftrightarrow (\neg Q \Rightarrow \neg P)$$

To decide the truth value of this messy proposition, we can evaluate subpropositions in a table and then combine their evaluations to evaluate the whole. If we do so, we determine that the column contains all T. That is, the expression is always true, regardless of the truth values of P and Q.

P	Q	$P \Rightarrow Q$	$\neg Q$	$\neg P$	$\neg Q \Rightarrow \neg P$	R
T	T	T	F	F	T	T
T	F	F	T	F	F	T
F	T	T	F	T	T	T
F	F	T	T	T	T	T

An expression (like this one) whose column contains all T is called a *tautology.*

5 Logical Systems

Now to proofs. A proof takes **axioms** and **definitions** and uses **deduction rules** to get desired conclusion.

5.1 Axioms

Definition An *axiom* is a proposition that is assumed to be true.

Axioms aren't proved—just accepted as reasonable assumptions. "Axiom" is Greek for "to think worthy," not "to be true". Here are some examples:

Axiom 1 If $a = b$ and $b = c$, then $a = c$.

This seems very reasonable! But sometimes the right choice of axiom is not clear.

Axiom 2 (Euclidean geometry) Given a line l and a point p not on l, there is exactly one line through p parallel to l.

Axiom 3 (Spherical geometry) Given a line l and a point p not on l, there is *no* line through p parallel to l.

Axiom 4 (Hyperbolic geometry) Given a line l and a point p not on l, there are *infinitely many* lines through p parallel to l.

No one of the three preceding axioms is better than the others; all yield equally good proofs. Of course, a different choice of axioms makes different propositions true. In fact, the axioms above are mutually contradictory—they can't all be true. Nevertheless, at different times you will want to work with different ones. For example, hyperbolic geometry is the right one to use if you stumble into a black hole...

Why do we have axioms? Without them, there is no place to start—no common ground on which to build proofs. Every argument has implicit assumptions. To be rigorous, it is important to state those assumptions explicitly as axioms.

Mathematicians still argue about which axioms are "right," e.g., there is a lot of debate about the Axiom of Choice (whose assertion and denial are equally consistent with other commonly agreed upon axioms).

Different axioms lead to different theorems. But anyone who accepts your axioms and deduction rules has to accept your theorems.

5.2 Definitions

Definitions are used to agree on what something means.

- A number is prime if no divisors other than itself
- A divisor is a number...
- A triangle is a polygon with 3 sides

Definitions are axiomatic: we can decide whatever we want, so long as we are explicit. They are generally built from prior definitions.

5.3 Deduction Rules

These are also called inference rules. They are rules for combining axioms and true propositions to construct more true propositions. A fundamental inference rule is *modus ponens*. This rule says that if p is true and $p \Rightarrow q$ is true, then q is true.

1. Modus ponens:
   Given that P and $P \Rightarrow Q$ are both true
   We can conclude: Q
2. Modus tolens (proof by contradiction):
   Given: $P \Rightarrow Q$, $\neg Q$
   We can conclude: $\neg P$
3. Syllogism:
   Given: $P \Rightarrow Q$, $Q \Rightarrow R$
   We can conclude: $P \Rightarrow R$

4. Given: $\neg P \Rightarrow \mathit{false}$:
   We can conclude: P
5. Given: P, $\neg P$:
   We can conclude: *false*
   (common way to get to previous rule: "proof by contradiction")
6. (Cases):
   Given: $Q \Rightarrow P$, $\neg Q \Rightarrow P$
   We can conclude: P.

The nice thing about these deduction rules is that they rely only on the truth values of the statements P, Q, R, and *not* on what the particular statements are. So we can apply them in many situations.

Truth tables provide a source of deduction rules; all such rules are accepted in mathematics.
Namely, if $P \Rightarrow Q$ is a tautology, then Given: P
Conclude: Q
is a valid inference rule.

For Example:

$$[(P \vee Q) \wedge \neg P] \implies Q$$

is a tautology, which leads to the following inference rule:

(Disjunctive Syllogism) Given: $P \vee Q$, $\neg P$
Conclude: Q

5.4 Formal proofs

In a complete, formal proof of a theorem, each step follows from previous steps using formal deduction rules. Such a process is ideally so systematic that it could be checked mechanically by a program. Mathematicians generally don't write proofs that formally. Rather, they write them in concise prose, but in such a way that others who are knowledgeable about proof methods can understand how to fill in the gaps.

Since proofs are supposed to be mechanically checkable, you might think of using a computer to produce proofs of true theorems automatically. Russel and Whitehead, in *Principia Mathematica*, tried to write down a sufficient set of axioms that would let us prove everything interesting. Then Hilbert posed (as one is his 23 famous problems) the problem of finding such a method. One of the great accomplishments of 20th century math is the proof that no such method exists (Gödel). In fact, we don't even have a proof that the axioms of math are consistent! Maybe they are all self-contradictory! But they have stood up well over time.

Even if we limit ourselves to things that can be proven, computers won't be much help. Finding a proof—even, say, just a proof that a given propositional logic statement is a tautology—is **NP-complete**. This means that it is intractable in a certain formal sense you will learn about in 6.045 or 6.046.

So we will always rely on human intuition to find the proofs, although we will be using computers more and more to check if they are right.

5.5 Proofs and the Real World

In the real world axioms capture our beliefs about what we are studying (e.g. Newton's mechanics). Definitions capture what we think is important (e.g. force, velocity). If we prove something with those axioms, we tend to believe it about the real world. If the real world turns out different, we need to change our axioms (Einstein made us discard Newton's axioms).

6 Propositional reasoning examples

Reading: Velleman Chapter 3, Rosen, Section 3.1

Some common types of proofs are direct proof, proof by contradiction, and proof by cases. Here, I'll say what these are and give some tiny examples. I'm not presenting formal logical proofs, but rather, proofs at about the level of detail commonly used by mathematicians. I'm using a pretty stylized, structured format, listing statements and reasons, as in high school geometry proofs. This lets me make uses of the proof rules explicit. They look a bit like programs.

In math books (including the ones in this course), proofs don't look like this. They are written in paragraphs, with the logical structure implicit in the prose. Caution: To really understand them, and write them yourself, you have to understand the logical structure. That's why I'm starting out by laying it out carefully.

Later in the course, as we become surer that you understand the logical structure, we will present many proofs in less structured styles. Paragraphs, semi-structured (laying out only the major steps). But ideally, it should be possible to rewrite any of these proofs in structured form (given enough time).

What about your own proofs? You can write them in structured form, paragraphs, or somewhere in between. Just make sure you get the proof logic right.

6.1 Direct proof

To prove that a proposition of the form $P \Rightarrow Q$ is true, it is enough to assume P and show Q. Written in a structured style, the pattern of such a proof is:

Prove: $P \Rightarrow Q$	
1. Assume P.	
2. Q	
3. QED	Implication

QED stands for Quod Erat Demonstrandum, Latin for "which was to be proved". Here, it's a shorthand for writing the goal out again as the final step.

Example 6.1 Let's prove that the sum of the first 100 numbers, $1 + ... + 100$, is 5050.

We'll rewrite this as an implication and use the direct proof rule above.

Prove: $S = 1 + ... + 100 \Rightarrow S = 5050$.	
1. Assume $S = 1 + ... + 100$.	
2. $S = 5050$	???
3. QED	Implication

Of course, the interesting part is showing step 2, where I've put the ???. Let's fill in those steps (calculations):

Prove: $S = 1 + ... + 100 \Rightarrow S = 5050$.	
1. Assume $S = 1 + ... + 100$.	
2. $S = 5050$	
1. $S = 100 + 99 + ... + 1$	Commutativity of addition
2. $2S = 101 + 101 + ... + 101$ (100 terms)	Add term-by-term
3. $2S = 10,100$	Add the terms up
4. QED	Divide by 2
3. QED	Implication

6.2 Proof by contradiction

To prove that a proposition P is true, it is enough to prove that $\neg P \Rightarrow \mathit{false}$.
The pattern is:

Prove: P	
1. $\neg P \Rightarrow \mathit{false}$	
2. QED	Contradiction

Some variants of this:

Prove: P
1. $\neg P \Rightarrow (Q \wedge \neg Q)$
2. QED Contradiction

Prove: P
1. $\neg P \Rightarrow Q$
2. $\neg Q$
3. QED Contradiction

Example 6.2

Assume: If Ricky gets at least six hours of sleep then he passes his 6042 final.
Ricky fails his 6042 final.
Prove: Ricky gets less than six hours of sleep.

Here, let $P =$ "Ricky gets less than six hours sleep.". $Q =$ "Ricky passes.". So the proof steps for proving P are:

1. If Ricky gets at least six hours of sleep then he passes. ($\neg P \Rightarrow Q$).
Assumed
2. Ricky fails ($\neg Q$). Assumed
3. QED Contradiction

6.3 Cases

To prove that a proposition P is true, it is enough to prove that Q and $\neg Q$ both imply P, where Q is any other proposition. Since Q is a proposition, it will be either true or false, and you have a proof for each case. This shows P is true in either case, so always true.

Prove: P
1. $Q \Rightarrow P$
2. $\neg Q \Rightarrow P$
3. QED Cases

Of course, you can generalize this to any number of predicates, provided that together they cover all possibilities. If $Q_1 \vee Q_2 \vee \ldots \vee Q_k$ is always true, and you can show each Q_i implies P, then you can infer P.

Example 6.3

Assume: $d =$ Britney's birth date (day of month, 1-31).
Prove: $d^2 + d$ is even.
1. $d = 1 \Rightarrow d^2 + d = 2$ is even.
2. $d = 2 \Rightarrow d^2 + d = 6$ is even.
3. $\ldots$
31. $d = 31 \Rightarrow d^2 + d = 992$ is even.
32. QED Cases

Of course we might imagine more clever proofs than this one, e.g., that avoid covering so many cases. For example, can we just case on whether d is even or odd?