Lecture 1: Introduction to Discrete Math and Logic

3 June 2019

Lecturer: J. Marcus Hughes

Content is borrowed from Susanna Epp's Discrete Mathematics with Applications and Andrew Altomare's notes.

1 Welcome

Hi! Welcome to Discrete Math this summer. I hope that we can learn a lot and have a great summer. I will try and make these lecture notes available before each class, but I will certainly make them available after the fact.

2 El Juego de las Amazonas

Instead of doing the boring thing of starting with the syllabus or just jumping into material, let's play a game. It is summer after all, the perfect time for games. **El Juego de las Amazonas** or **Game of the Amazons**, we'll call it **Amazons** for short, is a strategy game with simple rules but complex play. It was invented in 1988 by Walter Zamkauskas [1].

We will come back to this game throughout the semester at times as an example to think about. Amazons is played on a checker or chess board, but it can be any size. The designer played on a 10x10 board, but we'll play on 6x6 for a shorter game. For simplicity, we will also start with the Amazons in the corners. The rules are:

- The two players alternate turns. Play rock paper scissors or whatever other means you want to decide who goes first.
- The Amazon moves like a chess queen, any direction for as far as she wants except she cannot jump over pieces. We also don't capture pieces in Amazons.
- When she moves, she will shoot out an arrow (that moves like she does). You get to decide where the arrow lands. Wherever you stop it, that square is dead and neither player can place a piece there.
- Continues this way with players alternating until one of the players can no longer move. Whoever moved last wins!

As you play, get to know your opponent some. You can also think about some questions:

- How many possible games are there?

- How many possible positions are there?
- How many moves does the average game take?
- What kind of strategy exist in the game?
- What kind of symmetries exist in the game?
- What other variants of the game could you come up with that might be interesting?
- How would you mathematically describe this game?

3 Class Logistics

3.1 What is Discrete Math?

Why did we just play that game? It highlights some of the ideas of this course. In discrete math, we will learn about the mathematical formalism and proof writing skills that are the foundation of computer science and many other disciplines. Using such formalism, we can start to understand complex situations. For example, we might learn about how Amazons works and be able to create an amazing artificial intelligence to play against. Here are some more problems in discrete math:

- Route a packet reliably from one server to another on the internet (faster than existing protocols? even when routers can fail on us?)
- Search for web pages (better than Google, Bing or your favorite search engine??)
- Find the biggest clique on Facebook. How many people are in this clique? Who are they?
- Understand how to sequence the human genome.
- Find all the prime factors for really large number ($> 10^10000$).
- Write a program to play go (and play better than the best human champion?).
- Find anomalies in a data set and explain why they are anomalous (keeps your instructor awake at night!).

Discrete math is an umbrella term that collects many branches of math. While it does not have a formal definition, it focuses on things that are discrete instead of continuous. These terms have rigorous meaning, but you can think of discrete objects as ones with a bit of space around them instead of smoothly varying. For example, the integers 1 and 2 have a gap between them while the real numbers 1.1 and 1.2 have infinitely many numbers between them. Finite collections of things such as computers or people are always discrete!

We will cover many things in this class, but our emphasis is to provide an overview of many interesting discrete math areas so you are well versed in them for your future classes, interviews, and programming. I think you will find that the ideas are applicable in every day life. We will also emphasize proof writing and clear reasoning. You will likely be frustrated at times, but we learn most through challenges. Just take a breather or ask for help.

3.2 Who am I

I, Marcus, will be your instructor for the summer. I'm a PhD student. I grew up in Kentucky and lived there until 2014 when I graduated from the Carol Martin Gatton Academy of Mathematics and Science in Kentucky and moved to Massachusetts for my undergraduate degree at Williams College. I graduated with degrees in computer science and astronomy in 2018. I did consider a degree in astrophysics but realized I enjoyed the math and computation more. I then moved, here, to lovely Boulder and am now working toward my computer science PhD. I'm particularly excited by automating scientific discovery, with a current interest in anomaly detection as a means of guiding scientific inquiry by selecting interesting examples. I still have an interest in astronomical subjects and have focused more on solar phenomena as of late.

3.3 Policies

Most of the policies are outlined in the syllabus on our course GitHub page. If there are any significant changes, i.e. something more than just grammar, I will notify you through Piazza, our main channel for announcements, communication, and discussion this semester. I want to highlight a few things. First of all, any needs will be accommodated.

The course will follow the schedule available on GitHub. This tracks with previous courses and follows the textbook by Epp. I know that the textbook is expensive, but I strongly encourage you to find a copy of it somehow to use. You won't need it for homework, but you should read the book before class. I will post some goal questions that we aim to answer in each class. Even if you don't read Epp's book, you should read something to figure out how to answer them.

3.4 Philosophy

This course will likely not be easy for many of you. It's a different kind of class than you may be used to. We will focus on mathematical thinking with the goal of learning how to rigorously prove things. It's a critically important class and will help you in whatever endeavors you pursue. It will be easy to say get frustrated and declare "I'm not good at math." That reflects a very static mindset, assuming you only have so much capability. It makes you afraid of failure because failure confirms your fears of not being mathematically inclined.

With a growth mindset, I strongly believe that everyone can excel in this course. A growth mindset embraces mistakes as an opportunity to learn and reiterates that success comes from effort, not innate talent. Once you've grown accustomed to a static mindset, it will not be easy to switch to a growth mindset. But, we all can. I, personally, struggle to maintain a growth mindset but will do my best to encourage it. On each assignment, I will ask you to reflect what you think went well and what did not go well. This will both provide feedback for me to grow as a teacher and for you as a student to consciously adjust your approach. I will always encourage feedback on how the course is going and will openly provide you suggestions on how you're doing. Think of this class as an opportunity to *learn to think*, both in an organized, mathematical and a growing, non-static way.

4 Propositional Logic

4.1 Propositions

Welcome to our first unit, logic! There are many forms of logic: propositional logic, first-order logic, fuzzy logic, etc. We will begin with propositional logic. It won't help us determine the merit of an argument's content but it will help us identify structural errors and establish a framework to reason. Believe it or not, logic research is ongoing, both in its pure theoretical form and in application! Symbolic logic has provided the foundation for many computer science areas: circuit design, relational database theory, automata theory and computability, and artificial intelligence. We will talk some about these.

First, let's examine some arguments:

- **Argument 1** If the program syntax is faulty or if program execution results in division by zero, then the computer will generate an error message. Therefore, if the computer does not generate an error message, then the program syntax is correct and program execution does not result in division by zero.

- **Argument 2** If x is a real number such that $x < 2$ or $x > 2$, then $x^2 > 4$. Therefore, if $x^2 \not> 4$, then $x \not< 2$ and $x \not> 2$.

Notice how these arguments have very different content but a similar form. We use letters such as p, q, and r with symbols to abstract these content of an argument and represent its structure. In this case we might write "If p or q, then r. Therefore, if not r, then not p and not q."

Most of logic will just formalize your intuition. However, formal logic is much stricter about terms like *and*, *or*, and *statement* than in informal English. It will take some adjustment to use these terms in their formal sense and realize how imprecise everyday English is. For example, the basic building block of logic is a *proposition*, sometimes called a *statement*.

Definition: *Proposition/statement*
A **proposition** (sometimes called a **statement**) is a declarative statement that is either true or false, but not both.

This definition is very different than how we might use *statement* in common English, where it can be applied to many things. Here are some examples of propositions:

1. Boulder is a city in Colorado

2. $2 + 2 = 5$

3. $2 + 3 = 5$

Each of the statements above is either true or false and cannot be both. However, the statements below are **not** propositions.

1. How is the weather?: is not declarative
2. 6: does have a truth value
3. $x + 3 = 5$: depending on x it could be true or false, thus it can be both truth values!

For brevity we use a symbol to denote a proposition, conventional letters are $p, q, r, s, \ldots$

Exercise
Come up with some example propositions and some things that are not propositions. Explain why.

4.2 Compound statements

In English, we say much more complicated things. For example, I could say "Boulder and Denver are cities in Colorado but Boston is not; it is in Massachusetts." We can write this in our formal logic too! It becomes a compound statement because we combine propositions with operations such as negation ($\neg$), conjunction ($\wedge$), and disjunction ($\vee$).

4.2.1 Negation

Definition: *Negation*
Let p be a proposition. The *negation* of p, denoted $\neg p$, is the proposition "It is not the case that p". The truth value of $\neg p$ is the opposite of the truth value of p. You will sometimes see this denoted $\sim p$.

For example, let p="Andrew is wearing red shoes". Then $\neg p$ is the proposition "It is not the case that Andrew is wearing red shoes." We can summarize the possible truth values for various compound statements using truth tables. For example, below is the truth table for $\neg p$. We first consider all possible truth values for p and then iteratively build up the truth value for $\neg p$.

p	$\neg p$
T	F
F	T

Exercise
Prove that negating a negation, i.e. $\neg\neg p$, is the same as p. It's like saying "I am not not going to the store." You mean that you are going! Notice that in English there may be some implied connotation to using a double negative that does not exist in logical statements.

4.2.2 Conjunction and disjunction

We use conjunction to convey the idea of *and* in English and disjunction for *or*.

Definition: *Conjunction*
Let p and q be propositions. The *conjunction* of p and q, denoted by $p \wedge q$ is the proposition "p and q". The conjunction $p \wedge q$ has the truth value **T** if both p and q are **T** and is **F** otherwise.

Definition: *Disjunction*
Let p and q be propositions. The *disjunction* of p and q, denoted by $p \vee q$ is the proposition "p or q". The conjunction $p \vee q$ has the truth value **T** if either p or q is **T** and is **F** otherwise.

We can again summarize using a truth table.

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

Let p="It's dark outside" and q="My house is haunted" Then

1. "It is dark outside *and* my house is haunted" is $p \wedge q$
2. "It is light outside, but my house is still haunted" is $\neg p \wedge q$
3. "It is dark outside or my house is haunted" is $p \vee q$

Notice that some ideas can be tricky when translating to statements, even counterintuitive at first since they differ subtly from our colloquial English. Consider $a \leq x \leq b$ or "it is sunny and warm". What is the negation of these phrases? It might be tempting to say $a > x > b$ and "it is not sunny and warm" but these are wrong! The first statement literally means $a \leq x$ AND $x \leq b$ so the negation of this condition is that $a > x$ OR $x > b$, not necessarily both. The second statement's negation is ambiguous. Does it mean "it is not sunny and it is not warm" or does it mean "it is not sunny but it is warm?" English is not precise enough here. The negation is actually "is is not sunny or it is not warm" which we could write $\neg s \vee \neg w$.

Exercise
Are there any statements you could not say with just negation, conjunction, and disjunction?

4.2.3 Exclusive Or

Exclusive Or What about this one? “Karen can date Jeff or Ed". This is an example of *exclusive or* since it can’t be both.

Definition: *L*
et p and q be propositions. The *exclusive or* of p and q, denoted by $p \oplus q$ is the proposition that is true when exactly one of p or q is true and is false otherwise.

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

4.2.4 Conditional

Definition: *Conditional*
Let p and q be propositions. The *conditional* “if p then q", denoted by $p \to q$ is false when p is true but q is false, and true otherwise.

The conditional describes an if-then relationship between two propositions Think of the conditional $p \to q$ as defining a rule. What are the cases where the rule holds or where the rule is broken? For example. Let p=“It rains", q=“I bring an umbrella." Then the conditional statement $p \to q$ translates to “if [it rains] then [i bring an umbrella]". The only case that violates the rule is the case when it rains and I don’t bring an umbrella. Notice how when the condition p is violated and false, the result is always true. This is the vacuous case.

p	q	$p \to q$
T	T	T
T	F	F
F	T	T
F	F	T

Consider this example of translating phrases:

- I crash my bicycle only if it snows
- Define p=“it snows" and q=“I crash my bicycle"
- Then this statement can be expressed as $q \to p$

- Caution: “p only if q" is different from “p if q"

You will encounter many different ways of expressing the implication $p \to q$.

“if p, then q"	“p implies q"	“q if p"
“if p, q"	“p only if q"	“q whenever p"
“p is sufficient for q"	“a sufficient condition for q is p"	“q when p
“q is necessary for p"	“q follows from p"	“q unless not p"

There are three other conditionals closely related to $\boldsymbol{p} \to q$:

1. The *converse*: $q \to p$
2. The *inverse*: $\neg p \to \neg q$
3. The *contrapositive*: $\neg q \to \neg p$

4.2.5 Biconditional

Definition: *Biconditional*
Let p and q be propositions. The *biconditional* “p if and only if q", denoted by $p \leftrightarrow q$ is true when p and q have the same truth value, and false otherwise.

Example: A polygon is a triangle if and only if it has exactly 3 sides

- p=a polygon is a triangle
- q=a polygon has exactly 3 sides
- $p \leftrightarrow q$

p	q	$p \leftrightarrow q$
T	T	T
T	F	F
F	T	F
F	F	T

4.3 Truth Tables as tools

We can use truth tables to determine the truth values for compound statements. Consider $(p \to q) \wedge (q \to p)$. We can build this statement by examining each component with our known connectives.

p	q	$p \to q$	$q \to p$	$(p \to q) \wedge (q \to p)$
T	T	T	T	T
T	F	F	T	F
F	T	T	F	F
F	F	T	T	T

This should look familiar ($p \leftrightarrow q$)!

Another example: Suppose on are on an island where there are two types of people: *knights* that always tell the truth and *knaves* that always lie. Suppose on this island, you encounter two people, lets call them A and B. Suppose A tells you “I am a Knave or B is a Knight". Use a truth table to determine what kind of people A and B are.

- Let p=A is a Knight and q=B is a Knight
- A’s statement is then: $\neg p \vee q$
- How can we use a truth table to determine the nature of A and B?
  - We can test the compound proposition $\neg p \vee q \leftrightarrow p$ (if A is a Knight then his statement is honest, and if his statement was hones, that implies his Knighthood.)

p	q	$\neg p$	$\neg p \vee q$	$\neg p \vee q \leftrightarrow p$
T	T	F	T	T
T	F	F	F	F
F	T	T	T	F
F	F	T	T	F

$\Rightarrow$ A and B are both knights

Suppose instead that A tells you “B is a Knight" and B tells you “The two of us are of different types". Use a truth table to determine the sorts of people that A and B are.

- Using the same p, q as above, A says q and B says $p \oplus q$
- So A is a knight iff q and B is a knight iff $p \oplus q$
- Then test the proposition $(p \leftrightarrow q) \wedge (q \leftrightarrow p \oplus q)$

p	q	$p \leftrightarrow q$	$p \oplus q$	$p \oplus q \leftrightarrow q$	$(p \oplus q \leftrightarrow q) \wedge (p \leftrightarrow q)$
T	T	T	F	F	F
T	F	F	T	F	F
F	T	F	T	T	F
F	F	T	F	T	T

4.4 Satsifiability

Definition: *Satisfiabile*
A compound proposition is *satisfiable* if there is an assignment of truth values to its constituent propositions that makes it true. If there is no such case, then the compound proposition is unsatisfiable.

For example, $p \wedge \neg p$ is unsatisfiable.

Now, show that $(p \vee \neg q) \wedge (\neg p \vee q) \wedge (\neg p \vee \neg q)$ is satisfiable. We only need to demonstrate that there is one combination of truth values for p and q that makes this statement true. The first two conjuncts tell us that p and q must have the same truth values. The last one tells us that they must be **F**. Solution: p=**F**, q=**F** works so the compound proposition must be satisfiable

Now, show that $(p \rightarrow q) \wedge (p \rightarrow \neg q) \wedge (\neg p \rightarrow q) \wedge (\neg p \rightarrow \neg q)$ is unsatisfiable. To show that a compound proposition is unsatisfiable, we would need to demonstrate that for all combinations of truth values for p and q that makes this statement true. We can use a truth table or construct a logical argument.

The first two conjuncts tell us that p=**F**. If p=**F**, then the third conjunct tells us that q=**T**. But if p=**F** and q=**T** then the fourth construct is $\neg$**F** $\rightarrow$ $\neg$**T** or **T** $\rightarrow$ **F** which is **F**. Thus, a contradiction. So we conclude that the proposition must be unsatisfiable.

4.5 Necessary and sufficient conditions

Let n be a natural number $(0, 1, 2, 3, \ldots)$. It is *sufficient* that n be divisible by 12 for n to be divisible by 6. How can we represent this claim using a conditional?

Let r="n is divisible by 12" and s="n is divisible by 6." This statement is telling us that *under the condition that* n is divisible by 12, it must be divisible by 6. For a sufficient condition, the condition goes at the front of the conditional: $r \rightarrow s$.

It is *necessary* for warm surface air to start up convection in order for a severe summer thunderstorm to occur. How can we represent this claim using a conditional? Let t="severe summer thunderstorm occurs" and w="warm surface air spurs convection." This statement tells us that *under the condition that* a thunderstorm occurs, it must be the case that warm surface air has spurred convection. For a *necessary condition*, the condition goes at the end of the conditional: $t \rightarrow w$.

Consider the following statement: "If it snows, then I crash my bicycle riding home." Is it necessary or sufficient (or neither) ? It is sufficient. Similarly, "I crash my bicycle only if it snows." This is necessary. Considering both statements then we have $p \leftrightarrow q$. Note that we could do these example using truth tables, but they after are too large to be of any practical use. For n propositions, the truth table will have 2^n rows.

4.6 Sudoku satisfiability

Sudoku puzzles can be written (and solved) as satisfiability problems. It turns out that Sudoku puzzles would require 2^{729} rows, which is more than the number of atoms in the universe (between $2^{259} - 2^{272}$)

Let $p(i, j, n)$ represent the proposition that n occurs at row i and column j. There are 9 rows, 9 columns, and 9 numbers. Thus there are $9 \times 9 \times 9 = 729$ propositions. Hence, 2^{729} rows in your truth table... yikes!

Notation:

$$\bigwedge_{j=1}^{4} p_j = p_1 \wedge p_2 \wedge p_3 \wedge p_4$$

$$\bigvee_{j=1}^{4} p_j = p_1 \vee p_2 \vee p_3 \vee p_4$$

4.6.1 Each cell only contains one number:

$$\bigwedge_{i=1}^{9} \bigwedge_{j=1}^{9} \bigwedge_{n=1}^{9} \bigwedge_{m=1, m \neq n}^{9} p(i,j,n) \Rightarrow \neg p(i,j,m)$$

4.6.2 Row constraint:

- Row i contains a particular n:

$$\bigvee_{j=1}^{9} p(i,j,n)$$

- Row i contains all n:

$$\bigwedge_{n=1}^{9} \bigvee_{j=1}^{9} p(i,j,n)$$

- All rows contain all n:

$$\bigwedge_{i=1}^{9} \bigwedge_{n=1}^{9} \bigvee_{j=1}^{9} p(i,j,n)$$

4.6.3 Column constraint:

- Column j contains a particular n:

$$\bigvee_{i=1}^{9} p(i,j,n)$$

- Column j contains all n:

$$\bigwedge_{n=1}^{9} \bigvee_{i=1}^{9} p(i,j,n)$$

- All columns contain all n:

$$\bigwedge_{j=1}^{9} \bigwedge_{n=1}^{9} \bigvee_{i=1}^{9} p(i,j,n)$$

4.6.4 3×3 **squares constraint:**

- Let r indicate the row block and let c indicate the column block (0,1, or 2)
- Given r and c, how do we sum over the rows and columns within that 3×3 block?
- Each block spans rows 3r+1 to 3r+3 and 3c+1 to 3c+3, so
- For each block to contain $n = 1, 2, ..., 9$ we need
- $$\bigwedge_{r=0}^{2} \bigwedge_{c=0}^{2} \bigwedge_{n=1}^{9} \bigvee_{j=1}^{3} \bigvee_{i=1}^{3} p(i, j, n)$$

4.6.5 What we were given constraint:

$$p(1, 2, 4) \wedge p(2, 3, 5) \wedge \cdots \wedge p(9, 8, 7)$$

Chain these 5 constraints together with conjunctions and have a computer determine what unique set of truth values satisfy them!

References

[1] E. P. Jr. Amazons, 1999.