We will attempt to systemically answer the following question:
Does Δ imply α?

As well as the related questions:
Is Δ equivalent to α?
Is Δ satisfiable?
Is Δ valid?

We will discuss four major methods:

1. Enumerating Models
2. Inference Rules
3. Search (CSP/SAT)
4. NNF Circuits

(1) and (2) are deprecated but (2) is fundamental to (3) and (4).
This lecture will discuss the first two, and the next two will be discussed in the next.

Inference By Enumerating Models

Suppose our knowledge base $\Delta = \{A, A \vee B \implies C\}$.
Let $\alpha = C$; we can then ask: Does Δ imply α?

1. convert our knowledge base to CNF
$\Delta = A \wedge (A \vee B \implies C) = A \wedge (\neg(A \vee B) \vee C)$
2. build a truth table for the KB
3. use the models to solve:
$M(\Delta) = \{W1, W3\}$
$M(\alpha) = \{W1, W3, W5, W7\}$
4. If $M(\Delta) \subseteq M(\alpha)$, then $\Delta \models \alpha$

	A	B	C	Δ	α
W_1	T	T	T	T	T
W_2	T	T	F	F	F
W_3	T	F	T	T	T
W_4	T	F	F	F	F
W_5	F	T	T	F	T
W_6	F	T	F	F	F
W_7	F	F	T	F	T
W_8	F	F	F	F	F

Complexity: $O(2^n)$, n = # variables
(it can be used up to 25 variables!)
We can just as easily use this to solve the other questions.

Many logical systems demonstrate **monotonicity**.
$\equiv$ as the information increases, the set of entailed sentences can only grow.
For these problems, enumeration can cause the set to explode;
we need a model that can ignore irrelevant propositions.

Inference By Resolution

This method starts from the **Deduction Theorem**: Any question of implication can be converted into a SAT question.

If we wish to solve a SAT question, we can attempt to prove by contradiction. Say we want to prove that Δ implies α:

1. assume Δ & $\neg\alpha$
2. show that this causes a contradiction

To solve questions of satisfiability, we will use the concept of **inference rules**.
An inference rule is of the form $\frac{P_1, P_2}{P_3}$.
This says "if P1 and P2, then P3".

To apply proof by contradiction to questions of satisfiability, we:

1. convert the implication to normal form
2. apply inference rules until we either terminate or have a contradiction

If we can use rule R for a given derivation, we use the symbol $\vdash_R$.
If a rule will derive α from Δ any time $\Delta \not\models \alpha$, we call the rule **complete**.
Modus Ponens $\frac{A, A \implies B}{B}$ is not complete.
If a rule's denominator is always true provided the numerator, we call the rule **sound**.
$\frac{A, (A \vee B)}{(A \wedge B)}$ is not sound.

We would assume a sound and complete rule would be needed for proofs,
BUT our primary rule is actually not complete:

$$\text{Resolution: } \frac{A \lor B, \neg B \lor C}{A \lor C}$$

This may be counterintuitive, but we can see it holds:
$A \lor B = \neg A \implies B$
$\neg B \lor C = B \implies C$
$\neg A \implies B \;\&\&\; B \implies C = \neg A \implies C = A \lor C$
This is not complete, but it is **refutation complete** when applied to a CNF.
$\equiv$ given any CNF with a contradiction, it will derive the contradiction.

The Algorithm: We are asked the question "Does Δ imply α?"
This is equivalent to "Is $\Delta \land \neg\alpha$ a contradiction?".
We therefore apply resolution until either:

1. a contradiction occurs, in which case Δ implies α
2. no more resolution can be applied, in which case it does not

Example 1:
$\Delta : ((A \lor \neg B) \implies C) \land (C \implies D \lor \neg E) \land (E \lor D)$
$\alpha : A \implies D$
Does Δ imply α? $\rightarrow$ Is $\Delta \land \neg\alpha$ unsatisfiable?
The empty set cannot be true, so Δ implies α
Example 2:
$\Delta : ((A \land B) \implies C) \land A \land (C \implies D)$
$\alpha : C$
If we apply resolution, the algorithm will terminate; therefore, Δ does not imply α.

1. $\neg A \lor C$
2. $B \lor C$
3. $\neg C \lor D$
4. $E \lor D$
5. A
6. $\neg D$
7. $C < 0, 4 >$
8. $D \lor \neg E < 2, 6 >$
9. $E < 3, 5 >$
10. $D < 7, 8 >$
11. $\Phi < 5, 9 >$

This algorithm depends on a CNF, but how do we get one?

1. Get rid of all connections, but for $\{\lor \land \neg\}$
   - $A \iff B \rightarrow (A \implies B) \land (B \implies A)$
   - $A \implies B \rightarrow \neg A \lor B$
2. Use deMorgan's Law to push negatives inward
   - $\neg(A \land B) \rightarrow \neg A \lor \neg B$
   - $\neg(A \lor B) \rightarrow \neg A \land \neg B$
3. Distribute $\lor$ over $\land$
   - $(A \land B) \lor C \rightarrow (A \lor C) \land (B \lor C)$

Example: $A \iff (B \lor C)$

1. $\rightarrow (A \implies (B \lor C)) \land ((B \lor C) \implies A) \rightarrow (\neg A \lor B \lor C) \land (\neg(B \lor C) \lor A)$
2. $\rightarrow (A \lor B \lor C) \land ((\neg B \land \neg C) \lor A)$
3. $\rightarrow (\neg A \lor B \lor C) \land ((\neg B \lor A) \land (\neg C \lor A)$

Coincidentally, this is an NNF.