CSE 20
DISCRETE MATH

SPRING 2016

http://cseweb.ucsd.edu/classes/sp16/cse20-ac/

Reminders

- Exam 1 in one week
  - One note sheet ok
  - Review sessions Saturday / Sunday TBA
  - Assigned seats: seat map on Piazza shortly
- Homework 3 due Friday
- Office hours

Today's learning goals

- Relate boolean operations to applications

  Combinatorial circuits Logic puzzles
- Identify when and prove that a statement is a tautology or contradiction
- Prove propositional equivalences using truth tables
- Prove propositional equivalences using other known equivalences, e.g.
  - DeMorgan’s laws
  - Double negation laws
  - Distributive laws, etc.
- Compute the CNF and DNF of a given compound proposition.
- Identify when and prove that a statement is satisfiable or unsatisfiable, and when a set of statements is consistent or inconsistent.

Tautology and contradiction

Rosen p. 25

Tautology: compound proposition that is always T

Contradiction: compound proposition that is always F

p	q	F	T
T	T	F	T
T	F	F	T
F	T	F	T
F	F	F	T

Which of the following is a tautology?

A. p

B. $p \vee p$

C. $p \wedge p$

D. $p \vee \neg p$

E. $p \wedge \neg p$

CENTR101: CA   PCYNH109: AB

System specification + consistency

Rosen p. 23 #11

The router can send packets to the edge system only if it supports the new address space.

For the router to support the new address space, it is necessary that the latest software release be installed.

The router can send packets to the edge system if the latest software release is installed.

The router supports the new address space.

System specification + consistency

Rosen p. 23 #11

p only if q.

For q, it is necessary that r.

p if r.

q.

System specification + consistency

Rosen p. 23 #11

p only if q. $\neg q \rightarrow \neg p$

For q, it is necessary that r. $\neg r \rightarrow \neg q$

p if r. $r \rightarrow p$

q. q

System specification + consistency

Rosen p. 23 #11

p	q	r	$\neg q \rightarrow \neg p$			
T	T	T				
T	T	F				
T	F	T				
T	F	F				
F	T	T				
F	T	F				
F	F	T				
F	F	F				

How many rows will be set to F in 4th column?

A. 1
B. 2
C. 4
D. 8
E. None of the above

System specification + consistency

Rosen p. 23 #11

p	q	r	$\neg q \rightarrow \neg p$	$\neg r \rightarrow \neg q$	$r \rightarrow p$	q
T	T	T	T	T	T	T
T	T	F	T	F	T	T
T	F	T	F	T	T	F
T	F	F	F	T	T	F
F	T	T	T	T	F	T
F	T	F	T	F	T	T
F	F	T	T	T	F	F
F	F	F	T	T	T	F

System specification + consistency

Rosen p. 18

What property of the truth table guarantees consistency?

A. All output rows are T.
B. All output rows have both T and F.
C. At least one row has all outputs T.
D. At least one row has all outputs F.
E. None of the above

p	q	r	$\neg q \rightarrow \neg p$	$\neg r \rightarrow \neg q$	$r \rightarrow p$	q
			T	T	T	T
			T	F	T	T
			F	T	T	F
			F	T	T	F
			T	T	F	T
			T	F	T	T
F	F	T	T	T	F	F
F	F	F	T	T	T	F

System specifications are **consistent** if they do not contain conflicting requirements

Puzzle: Muddy children

Rosen Example 8, page 20

- Two kids play in the mud outside. When they are done, their parent says "at least one of you has a muddy forehead." The children can't see themselves but can see each other. In what situation can each child deduce whether s/he has a muddy forehead?

A. Both of them have muddy foreheads.
B. Neither of them has a muddy forehead.
C. Exactly one of the has muddy forehead.
D. None of the above – they can't be sure.
E. I don't know.

Puzzle: Muddy children

Rosen Example 8, page 20

- Two kids play in the mud outside. When they are done, their parent says "at least one of you has a muddy forehead." The children can't see themselves but can see each other. In the situation where they both have muddy foreheads, will any of the following pieces of information help them deduce the truth?

A. They see each other's muddy foreheads.
B. They hear each other's answer that they don't know whether they have a muddy forehead.
C. They hear each other's answer that they each know that they have mud on their forhead.
D. None of the above – they can't be sure.
E. I don't know.

Logical equivalences

Rosen p. 25

Compound propositions that have the same truth values in all possible cases are **logically equivalent**, denoted $\equiv$.

p	q	
T	T	?
T	F	?
F	T	?
F	F	?

De Morgan Laws

Rosen p. 26

$$\neg(p \land q) \equiv \neg p \lor \neg q$$

$$\neg(p \lor q) \equiv \neg p \land \neg q$$

Replacing the main connective!

(Some) Useful equivalences

Rosen p. 26-28

$$\neg(p \wedge q) \equiv \neg p \vee \neg q$$

$$\neg(p \vee q) \equiv \neg p \wedge \neg q$$

$$p \vee q \equiv q \vee p$$

$$p \wedge q \equiv q \wedge p$$

$$p \wedge F \equiv F$$

$$p \vee T \equiv T$$

$$p \wedge T \equiv p$$

$$p \vee F \equiv p$$

$$p \rightarrow q \equiv \neg p \vee q$$

$$p \rightarrow q \equiv \neg q \rightarrow \neg p$$

…. 32 equivalences listed in book!

Can replace *p* and *q* with any (compound) proposition

(Some) Useful equivalences

Rosen p. 26-28

- For constructing (minimal) circuits with specified gates
  - only NOTs?
  - only ANDs?

$p \wedge T \equiv p$

$p \vee F \equiv p$

…. 32 equivalences listed in book!

Can replace *p* and *q* with any (compound) proposition

(Some) Useful equivalences

Rosen p. 26-28

- For simplying and evaluating complicated compound propositions
  - Remove parentheses?
  - Reduce subexpressions to simpler ones

…. 32 equivalences listed in book!

Can replace *p* and *q* with any (compound) proposition

(Some) Useful equivalences

Rosen p. 26-28

- For devising proofs of statements
    - Translate using existing logical structure.
    - Try to apply known proof strategy.
    - Rewrite in equivalent way to apply additional proof strategies.

(more on this later)

Can replace *p* and *q* with any (compound) proposition

Sample equivalence proof

- Prove that $(p \wedge q) \rightarrow r$ is logically equivalent to $p \rightarrow (q \rightarrow r)$

Are these compound propositions logically equivalent to $(p \rightarrow q) \rightarrow r$ *?*

Other laws of equivalence

Rosen p. 29-31

Any compound proposition can be translated to one using …

A. only ANDs and ORs.
B. only ANDs and NOTs.
C. only IFs.
D. only NOTs.
E. None of the above

Other laws of equivalence

Rosen p. 35 #42-53

Any compound proposition can be translated to one using …

A. only ANDs and ORs.
B. only ANDs and NOTs.
C. only IFs.
D. only NOTs.
E. None of the above

Functionally complete collection of connectives.

Functionally complete set of connectives *Rosen p. 35 #42-53*

Rewriting compound propositions using only NOT, AND

1. Work from the inside out, starting with connectives applied to basic propositions / propositional variables.

Functionally complete set of connectives *Rosen p. 35 #42-53*

Rewriting compound propositions using only NOT, AND

1. Work from the inside out …
2. For each connective, replace it with an equivalent form that uses only NOT, AND:
   - If the connective is NOT or AND, do nothing.

Functionally complete set of connectives *Rosen p. 35 #42-53*

Rewriting compound propositions using only NOT, AND

1. Work from the inside out …
2. For each connective, replace it with an equivalent form that uses only NOT, AND:
   - If the connective is NOT or AND, do nothing.
   - If the connective is OR: replace $p \vee q$ with …

Functionally complete set of connectives *Rosen p. 35 #42-53*

Rewriting compound propositions using only NOT, AND

1. Work from the inside out …
2. For each connective, replace it with an equivalent form that uses only NOT, AND:
   - If the connective is NOT or AND, do nothing.
   - If the connective is OR: replace $p \vee q$ with … $\neg(\neg p \wedge \neg q)$

Functionally complete set of connectives *Rosen p. 35 #42-53*

Rewriting compound propositions using only NOT, AND

1. Work from the inside out …
2. For each connective, replace it with an equivalent form that uses only NOT, AND:
   - If the connective is NOT or AND, do nothing.
   - If the connective is OR: replace $p \vee q$ with … $\neg(\neg p \wedge \neg q)$
   - If the connective is IF..THEN: replace $p \rightarrow q$ with …

Functionally complete set of connectives *Rosen p. 35 #42-53*

Rewriting compound propositions using only NOT, AND

1. Work from the inside out …
2. For each connective, replace it with an equivalent form that uses only NOT, AND:
   - If the connective is NOT or AND, do nothing.
   - If the connective is OR: replace $p \vee q$ with … $\neg(\neg p \wedge \neg q)$
   - If the connective is IF..THEN: replace $p \rightarrow q$ with … $\neg(p \wedge \neg q)$

Functionally complete set of connectives *Rosen p. 35 #42-53*

Rewriting compound propositions using only NOT, AND

1. Work from the inside out …
2. For each connective, replace it with an equivalent form that uses only NOT, AND:
   - If the connective is NOT or AND, do nothing.
   - If the connective is OR: replace $p \vee q$ with … $\neg(\neg p \wedge \neg q)$
   - If the connective is IF..THEN: replace $p \rightarrow q$ with … $\neg(p \wedge \neg q)$
   - If the connective is IFF: replace $p \leftrightarrow q$ with …

Functionally complete set of connectives *Rosen p. 35 #42-53*

Rewriting compound propositions using only NOT, AND

1. Work from the inside out …
2. For each connective, replace it with an equivalent form that uses only NOT, AND:
   - If the connective is NOT or AND, do nothing.
   - If the connective is OR: replace $p \vee q$ with … $\neg(\neg p \wedge \neg q)$
   - If the connective is IF..THEN: replace $p \to q$ with … $\neg(p \wedge \neg q)$
   - If the connective is IFF: replac $p \leftrightarrow q$ with … $\neg(p \wedge \neg q) \wedge \neg(\neg p \wedge q)$

Functionally complete set of connectives *Rosen p. 35 #42-53*

Rewriting compound propositions using only NOT, AND

1. Work from the inside out …
2. For each connective, replace it with an equivalent form that uses only NOT, AND:
   - If the connective is NOT or AND, do nothing.
   - If the connective is OR: replace $p \vee q$ with … $\neg(\neg p \wedge \neg q)$
   - If the connective is IF..THEN: replace $p \rightarrow q$ with … $\neg(p \wedge \neg q)$
   - If the connective is IFF: replace $p \leftrightarrow q$ with … $\neg(p \wedge \neg q) \wedge \neg(\neg p \wedge q)$
   - If the connective is XOR: replace $p \oplus q$ with …

Functionally complete set of connectives *Rosen p. 35 #42-53*

Rewriting compound propositions using only NOT, AND

1. Work from the inside out …
2. For each connective, replace it with an equivalent form that uses only NOT, AND:
   - If the connective is NOT or AND, do nothing.
   - If the connective is OR: replace $p \vee q$ with … $\neg(\neg p \wedge \neg q)$
   - If the connective is IF..THEN: replace $p \rightarrow q$ with … $\neg(p \wedge \neg q)$
   - If the connective is IFF: replace $p \leftrightarrow q$ with … $\neg(p \wedge \neg q) \wedge \neg(\neg p \wedge q)$
   - If the connective is XOR: replace $p \oplus q$ with … $\neg(\neg(p \wedge \neg q) \wedge \neg(\neg p \wedge q))$

Functionally complete set of connectives *Rosen p. 35 #42-53*

Example: express $A \to (B \vee C)$ as a logically equivalent compound proposition that only uses ANDs and NOTs.

Functionally complete set of connectives *Rosen p. 35 #42-53*

Example: express $A \to (B \vee C)$ as a logically equivalent compound proposition that only uses ANDs and NOTs.

Use $p \vee q \equiv \neg(\neg p \wedge \neg q)$ to rewrite intermediate step:

$$A \to \neg(\neg B \wedge \neg C)$$

Functionally complete set of connectives Rosen p. 35 #42-53

Example: express $A \to (B \vee C)$ as a logically equivalent compound proposition that only uses ANDs and NOTs.

Use $p \vee q \equiv \neg(\neg p \wedge \neg q)$ to rewrite intermediate step:

$$A \to \neg(\neg B \wedge \neg C)$$

Use $p \to q \equiv \neg(p \wedge \neg q)$ to rewrite:

$$\neg(A \wedge \neg(\neg(\neg B \wedge \neg C)))$$

Simplify double negation:

$$\neg(A \wedge \neg B \wedge \neg C)$$

Sample questions

Given compound proposition

- What is its main connective?
- Is it a tautology? a contradiction?
- Construct its truth table.
- Construct a combinatorial circuit using …. that produces the values of this proposition as output.
- Is it logically equivalent to …. ?
- Find its negation and "simplify".
- Find a logically equivalent compound proposition that …

Going backwards

Given compound proposition, use

- Truth tables
- Logical equivalences

to compute truth values.

Reverse?

Given truth table settings, want compound proposition with that output.

CNF and DNF

Rosen p. 35 #42-53

Conjunctive normal form: AND of ORs (of variables or their negations).

Disjunctive normal form: OR of ANDs (of variables or their negations).

Which of the following is in CNF?

A. $p \vee q$
B. $\neg(p \vee q)$
C. $(\neg p \vee q) \wedge (p \vee \neg q)$
D. $(p \wedge q) \vee (\neg p \wedge \neg q)$
E. More than one of the above.

Reverse-engineering

p	*q*	*r*	*?*
T	T	T	T
T	T	F	T
T	F	T	F
T	F	F	T
F	T	T	F
F	T	F	F
F	F	T	T
F	F	F	F

Reverse-engineering

Approach 1: classify rows based on one variable

p	*q*	*r*	*?*
T	T	T	T
T	T	F	T
T	F	T	F
T	F	F	T
F	T	T	F
F	T	F	F
F	F	T	T
F	F	F	F

$$p \wedge (r \rightarrow q)$$

$$\neg p \wedge \neg q \wedge r$$

Reverse-engineering

Approach 2: algorithmically convert to normal form

p	*q*	*r*	*?*
T	T	T	T
T	T	F	T
T	F	T	F
T	F	F	T
F	T	T	F
F	T	F	F
F	F	T	T
F	F	F	F

Reverse-engineering

Approach 2: algorithmically convert to normal form

DNF: when is output T?

p	*q*	*r*	*?*
T	T	T	T
T	T	F	T
T	F	T	F
T	F	F	T
F	T	T	F
F	T	F	F
F	F	T	T
F	F	F	F

LAND IN THESE ROWS!

Reverse-engineering

Approach 2: algorithmically convert to normal form

DNF: when is output T?

p	*q*	*r*	*?*
T	T	T	T
T	T	F	T
T	F	T	F
T	F	F	T
F	T	T	F
F	T	F	F
F	F	T	T
F	F	F	F

Reverse-engineering

Approach 2: algorithmically convert to normal form

DNF: when is output T?

p	*q*	*r*	*?*	
T	T	T	T	$p \land q \land r$
T	T	F	T	$p \land q \land \neg r$
T	F	T	F	
T	F	F	T	$p \land \neg q \land \neg r$
F	T	T	F	
F	T	F	F	
F	F	T	T	$\neg p \land \neg q \land r$
F	F	F	F	

Reverse-engineering

Approach 2: algorithmically convert to normal form

DNF: when is output T?

p	q	r	?
T	T	T	T
T	T	F	T
T	F	T	F
T	F	F	T
F	T	T	F
F	T	F	F
F	F	T	T
F	F	F	F

$p \land q \land r$

$p \land q \land \neg r$

$p \land \neg q \land \neg r$

$\neg p \land \neg q \land r$

$$(p \land q \land r) \lor (p \land q \land \neg r) \lor (p \land \neg q \land \neg r) \lor (\neg p \land \neg q \land r)$$

Reverse-engineering

Approach 2: algorithmically convert to normal form

DNF: when is output T?

p	*q*	*r*	*?*
T	T	T	T
T	T	F	T
T	F	T	F
T	F	F	T
F	T	T	F
F	T	F	F
F	F	T	T
F	F	F	F

Why is it ok to ignore the F rows?

A. It's not, we need to add clauses from them too.
B. If we don't specify a value it's automatically F.
C. The definition "and" will set the output to "F" if the T row conditions aren't met.
D. More than one of the above.
E. None of the above.

Reverse-engineering

Approach 2: algorithmically convert to normal form

CNF: when is output F?

p	*q*	*r*	*?*
T	T	T	T
T	T	F	T
T	F	T	F
T	F	F	T
F	T	T	F
F	T	F	F
F	F	T	T
F	F	F	F

AVOID THESE ROWS!

Reverse-engineering

Approach 2: algorithmically convert to normal form

CNF: when is output F?

p	q	r	?
T	T	T	T
T	T	F	T
T	F	T	F
T	F	F	T
F	T	T	F
F	T	F	F
F	F	T	T
F	F	F	F

AVOID THESE ROWS!

$$\neg(\neg p \land \neg q \land \neg r) \equiv$$
$$p \lor q \lor r$$

Reverse-engineering

Approach 2: algorithmically convert to normal form

CNF: when is output F?

p	*q*	*r*	*?*	
T	T	T	T	
T	T	F	T	
T	F	T	F	$\neg p \lor q \lor \neg r$
T	F	F	T	
F	T	T	F	$p \lor \neg q \lor \neg r$
F	T	F	F	$p \lor \neg q \lor r$
F	F	T	T	
F	F	F	F	$p \lor q \lor r$

Reverse-engineering

Approach 2: algorithmically convert to normal form

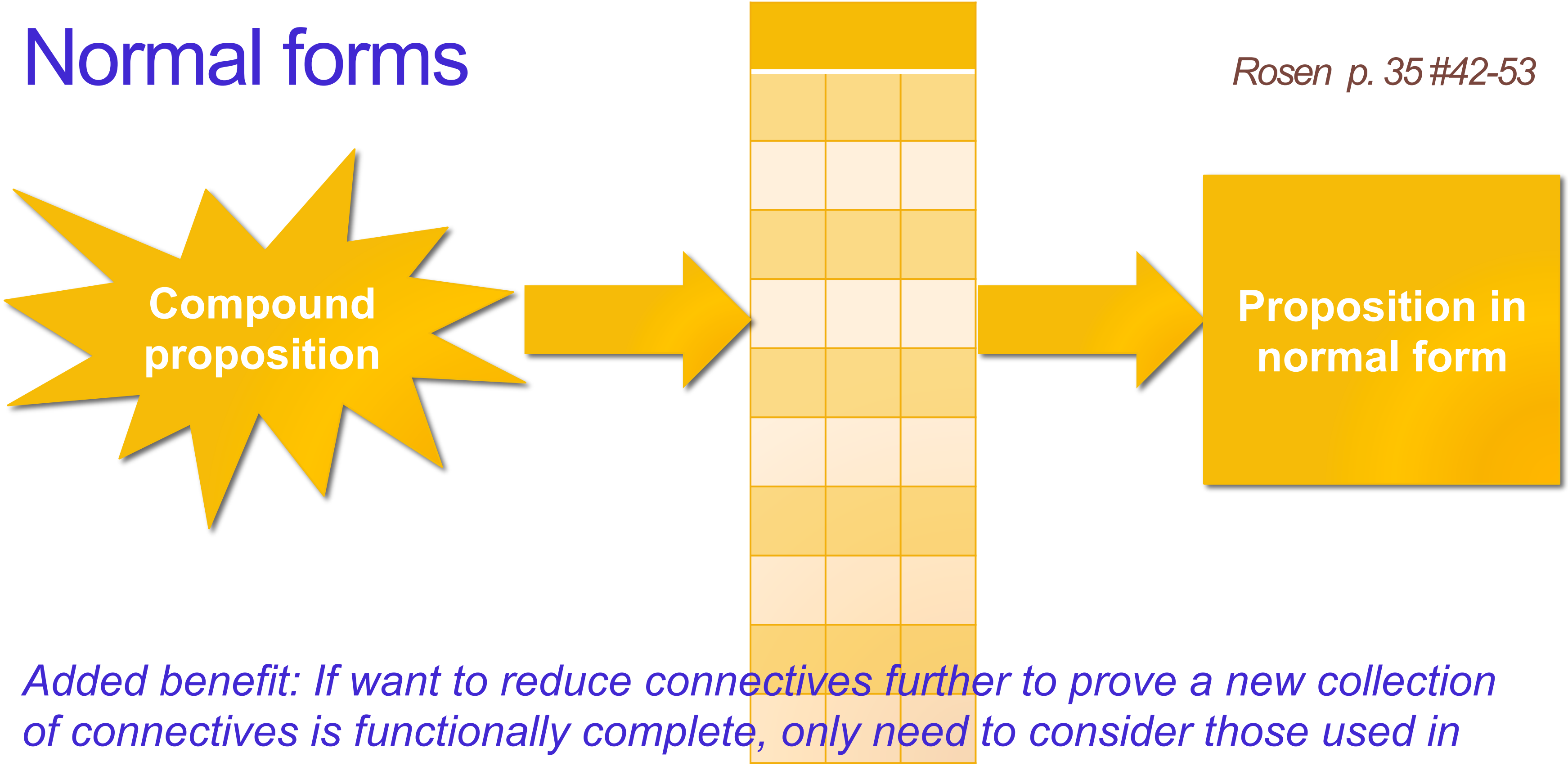
CNF: when is output F?

p	q	r	?	
T	T	T	T	
T	T	F	T	
T	F	T	F	$\neg p \vee q \vee \neg r$
T	F	F	T	
F	T	T	F	$p \vee \neg q \vee \neg r$
F	T	F	F	$p \vee \neg q \vee r$
F	F	T	T	
F	F	F	F	$p \vee q \vee r$

$$(\neg p \vee q \vee \neg r) \wedge (p \vee \neg q \vee \neg r) \wedge (p \vee \neg q \vee r) \wedge (p \vee q \vee r)$$

Normal forms

Rosen p. 35 #42-53



Added benefit: If want to reduce connectives further to prove a new collection of connectives is functionally complete, only need to consider those used in normal form.

Reminders

- Exam 1 in one week
  - One note sheet ok
  - Review sessions Saturday / Sunday
  - Assigned seats: seat map on Piazza shortly
- Homework 3 due Friday
- Office hours