CSE 20
DISCRETE MATH

WINTER 2017

http://cseweb.ucsd.edu/classes/wi17/cse20-ab/

Reminders

- Homework 3 due Sunday at noon
- Exam 1 in one week
  - One note card can be used. Bring photo ID.
  - Review sessions
    - Thursday in class
    - Saturday January 28 11am-12:50pm PETER108 Recommended textbook qs
    - Monday January 30 8pm-9:50pm SOLIS107 Exam Practice Sheet
  - Assigned seats: seat map on Piazza shortly

- Office hours

Today's learning goals

- 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.
- Relate boolean operations to applications

Combinatorial circuits

Logic puzzles

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. $p \wedge \neg p$

Clicker frequency: AC

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	?

Notice:
A and B are logically equivalent
iff
A←→B is a tautology

De Morgan Laws

Rosen p. 26

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

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

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 $(p \rightarrow q) \rightarrow r$ *?*
You'll decide in HW3!

Other laws of equivalence

Rosen p. 29-31

Any compound proposition can be translated to one using …

A. only ANDs.
B. only ORs.
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.
B. only ORs.
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*

Claim: The connectives AND, NOT are functionally complete.

Any compound proposition can be rewritten as a logically equivalent one that only has the operators AND, NOT

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

Claim: The connectives AND, NOT are functionally complete.

Any compound proposition can be rewritten as a logically equivalent one that only has the operators AND, NOT

Example: $p \wedge (q \wedge \neg r)$

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

Claim: The connectives AND, NOT are functionally complete.

Any compound proposition can be rewritten as a logically equivalent one that only has the operators AND, NOT

Example: $p \vee q$

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

Claim: The connectives AND, NOT are functionally complete.

Any compound proposition can be rewritten as a logically equivalent one that only has the operators AND, NOT

Example: $p \to q$

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

Claim: The connectives AND, NOT are functionally complete.

Any compound proposition can be rewritten as a logically equivalent one that only has the operators AND, NOT

Example: $p \leftrightarrow q$

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

Claim: The connectives AND, NOT are functionally complete.

Any compound proposition can be rewritten as a logically equivalent one that only has the operators AND, NOT

Example: $p \oplus q$

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 $p \to q$ 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 $A \rightarrow (B \vee C)$ 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 and use associativity: $\neg(A \wedge \neg B \wedge \neg C)$

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. *E.g. Think back to HW2 Q5*

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.

In this context, the propositional variable A can be interpreted as an

- AND (of itself), and as an
- OR (of itself)

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

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

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

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	$\neg p \vee q \vee \neg r$
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	$p \vee q \vee r$

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

$\neg p \vee q \vee \neg r$

$p \vee \neg q \vee \neg r$

$p \vee \neg q \vee r$

$p \vee q \vee r$

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

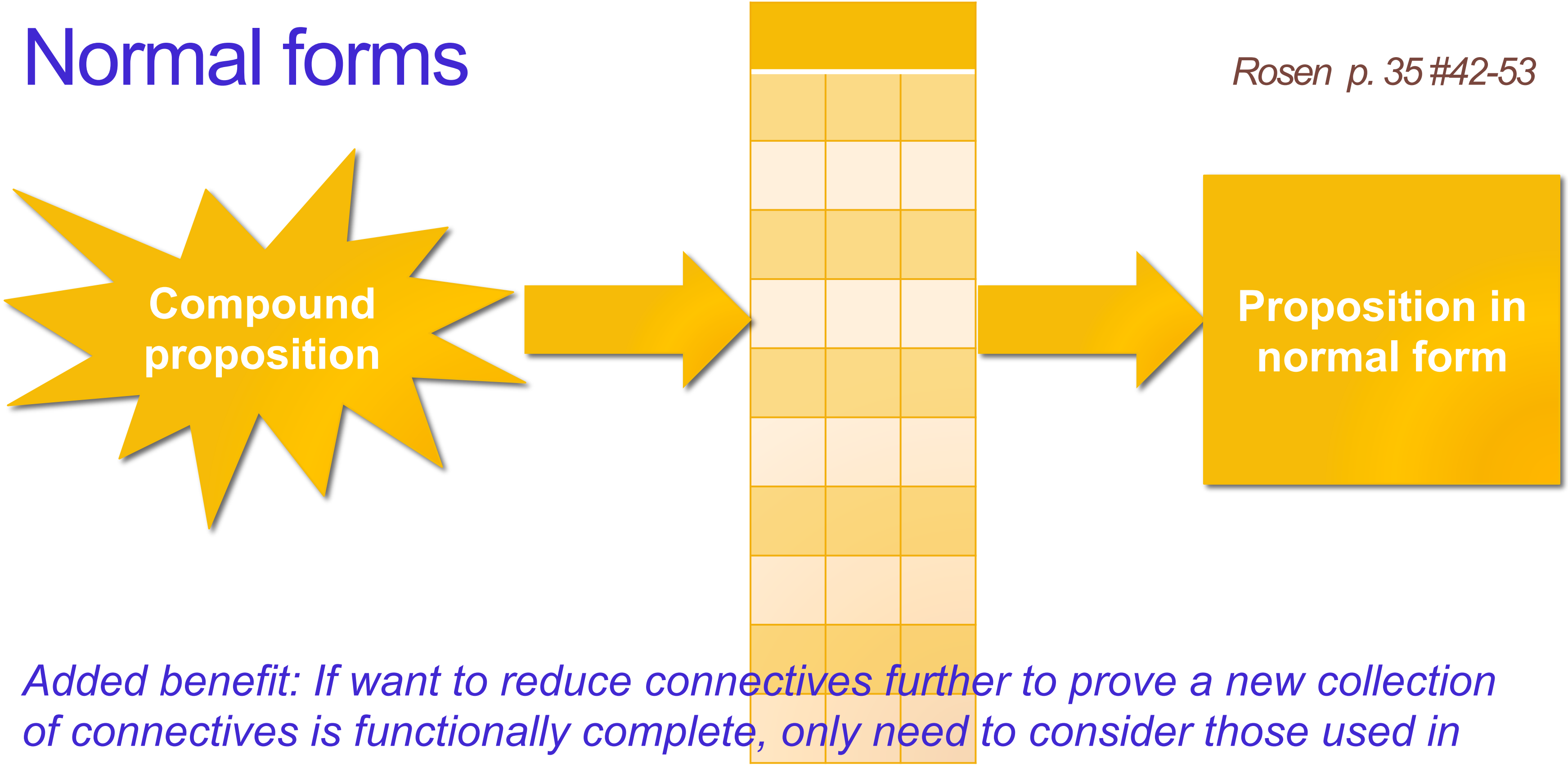
Payoff

- Any output column of a truth table (assignment of T/F to each combination of T/F input values) can be realized as a compound proposition.

- The collection $\vee$ $\wedge$ $\neg$ is functionally complete.

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.

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. If the older child sees mud on the younger child's forehead …

A. The older child knows both of them have muddy foreheads.
B. The older child knows the younger child has a muddy forehead.
C. The older child knows that they themselves have a 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. If the older child (even being a perfect reasoner) doesn't have enough information to tell whether s/he has mud on their forehead…

A. The younger child knows the older one has muddy forehead.
B. The younger child knows the older one has no mud.
C. The younger child knows s/he himself has no mud.
D. None of the above – they can't be sure.
E. I don't know.

Reminders

- Homework 3 due Sunday at noon

- Exam 1 in one week
  - One note card can be used. Bring IDs.
  - Review sessions
    - In class Thursday
    - On the weekend TBA (if we get a lecture hall)
    - Monday January 30 8pm-9:50pm
  - Assigned seats: seat map on Piazza shortly

- Office hours listed on class calendar