College of Engineering & Applied Sciences

CSPB 2824

Discrete Sturctures

Propositional Logic Mastery Workbook

Taylor Larrechea

2023

Sections

Mastery Workbook 2

Propositional Logic Workbook

I have neither given nor received unauthorized assistance.

Taylor James Larrechea

Problem 1

Problem Statement

Before anything else, **annotate** examples 7 and 8 from page 30. You **MUST** use this **EXACT** format when showing logical equivalences (and proofs) to get credit for your work in this course.

Annotations can be seen on the next page.

logical equivalences, using one of the equivalences in Table 6 at a time, starting with $\neg(p \to q)$ and ending with $p \wedge \neg q$. We have the following equivalences.

$$
\begin{aligned}
\neg(p \to q) &\equiv \neg(\neg p \vee q) && \text{by Example 3}\\
&\equiv \neg(\neg p) \wedge \neg q && \text{by the second De Morgan law}\\
&\equiv p \wedge \neg q && \text{by the double negation law}
\end{aligned}
$$

◀

EXAMPLE 7 Show that $\neg(p \vee (\neg p \wedge q))$ and $\neg p \wedge \neg q$ are logically equivalent by developing a series of logical equivalences.

↳ Kinda like equality

Solution: We will use one of the equivalences in Table 6 at a time, starting with $\neg(p \vee (\neg p \wedge q))$ and ending with $\neg p \wedge \neg q$. (*Note:* we could also easily establish this equivalence using a truth table.) We have the following equivalences.

Use properties from table 6 to prove things

→ Flips disj → conj, distributes ¬

$$
\begin{aligned}
\neg(p \vee (\neg p \wedge q)) &\equiv \neg p \wedge \neg(\neg p \wedge q) && \text{by the second De Morgan law}\\
&\equiv \neg p \wedge [\neg(\neg p) \vee \neg q] && \text{by the first De Morgan law}\\
&\equiv \neg p \wedge (p \vee \neg q) && \text{by the double negation law}\\
&\equiv (\neg p \wedge p) \vee (\neg p \wedge \neg q) && \text{by the second distributive law}\\
&\equiv \mathbf{F} \vee (\neg p \wedge \neg q) && \text{because } \neg p \wedge p \equiv \mathbf{F}\\
&\equiv (\neg p \wedge \neg q) \vee \mathbf{F} && \text{by the commutative law for disjunction}\\
&\equiv \neg p \wedge \neg q && \text{by the identity law for } \mathbf{F}
\end{aligned}
$$

↳ Original Exp.

→ De Morgan applied again

→ $\neg(\neg P)$ changes to P

→ Acts similar to reg distr

→ Impossible for $\neg x \wedge x \equiv T$

→ Kinda like $P \vee q \equiv q \vee P$

→ "F" not important here

This example is eerily similar to solving systems of eqn. i.e.

$x + 2y = 3$, $x + y = 0$ } $y = x - 0$, $x + 2(x - 0) = 3$, $3x = 3$, $x = 1$, $y = -1$

Consequently $\neg(p \vee (\neg p \wedge q))$ and $\neg p \wedge \neg q$ are logically equivalent. ◀

↳ Kinda like saying QED or □

EXAMPLE 8 Show that $(p \wedge q) \to (p \vee q)$ is a tautology.

↳ use RBI ↳ Always true

Solution: To show that this statement is a tautology, we will use logical equivalences to demonstrate that it is logically equivalent to **T**. (*Note:* This could also be done using a truth table.)

→ using past result from T table

$$
\begin{aligned}
(p \wedge q) \to (p \vee q) &\equiv \neg(p \wedge q) \vee (p \vee q) && \text{by Example 3}\\
&\equiv (\neg p \vee \neg q) \vee (p \vee q) && \text{by the first De Morgan law}\\
&\equiv (\neg p \vee p) \vee (\neg q \vee q) && \text{by the associative and commutative laws for disjunction}\\
&\equiv \mathbf{T} \vee \mathbf{T} && \text{by Example 1 and the commutative law for disjunction}\\
&\equiv \mathbf{T} && \text{by the domination law}
\end{aligned}
$$

→ Flipping conj to disj

→ All disj's, order doesn't matter

Tautological

Always true :)

Take away:
- All disj's → re-arrange!
- All conj's → re-arrange!
- De Morgans is our friend
- RBI used most probs

◀

Propositional Satisfiability

A compound proposition is **satisfiable** if there is an assignment of truth values to its variables that makes it true. When no such assignments exists, that is, when the compound proposition is false for all assignments of truth values to its variables, the compound proposition is **unsatisfiable**. Note that a compound proposition is unsatisfiable if and only if its negation is true for all assignments of truth values to the variables, that is, if and only if its negation is a tautology.

When we find a particular assignment of truth values that makes a compound proposition true, we have shown that it is satisfiable; such an assignment is called a **solution** of this particular

Problem 1 Summary

Procedure

- Provide annotations for a proof format that is to be used in this class

Key Concepts

- This problem involves the proper format that should be used for proofs in this class
- Always begin the proof with one side of the theorem, but **never** start a proof with what is to be proven
- Comments should be added to the right for each line of a proof

Variations

- Because this problem involves annotating an example, a similar problem would be annotating a different problem
  - In this case we would use the same procedure that was used in the original problem

Problem 2

Problem Statement

We will prove the first 5 logical equivalences in Chart 7 page 28.

- USE only RESULTS FROM **Chart 2 and Chart 6** to justify steps.
- When making logical chains of equivalences use the REQUIRED format.
- Prove these in this order and **you may use the results as you go.**

The logical equivalences that must be proven are:

(a) This first one is what Chris calls RBI and it is essential because it converts a conditional into a disjunction. **Prove this with just a truth table.**

$$p \to q \equiv \neg p \vee q$$

(b) Carefully repeat the proof of the following using the method without a truth table from the video. Take note of what tricks Chris uses - as we will need these as we go. (Challenge - can you prove it another way?)

$$p \to q \equiv \neg q \to \neg p$$

(c) Third one (hint: RBI works both ways).

$$p \vee q \equiv \neg p \to q$$

(d) Fourth one, (think about which rule 'flips' conjunctions and disjunctions).

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

(e) Finally, for the fifth one, see if you can do it on your own.

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

Solution - Part (a)

We will first prove $p \to q \equiv \neg p \vee q$ with the use of a truth table.

p	**q**	$\neg$**p**	**p** $\to$ **q**	$\neg$**p** $\vee$ **q**
T	T	F	T	T
T	F	F	F	F
F	T	T	T	T
F	F	T	T	T

Consequently, $p \to q$ is logically equivalent to $\neg p \vee q$.

Solution - Part (b)

Starting with RBI, we can say

$$\begin{aligned}
p \to q &\equiv \neg p \vee q && \text{(RBI)}\\
&\equiv q \vee \neg p && \text{(Commutative Property)}\\
&\equiv \neg q \vee \neg(\neg p) && \text{(Double Negation)}\\
&\equiv \neg q \vee p && \text{(Double Negation Law)}\\
&\equiv \neg q \to \neg p && \text{(RBI)}
\end{aligned}$$

Consequently, $p \to q$ is logically equivalent to $\neg q \to \neg p$.

Solution - Part (c)

Starting with RBI, we can say

$$\begin{aligned} \neg p \to q &\equiv \neg(\neg p) \vee q && \text{(RBI)} \\ &\equiv p \vee q && \text{(Double Negation Law)} \end{aligned}$$

Consequently, $p \vee q$ and $\neg p \to q$ are logically equivalent.

Solution - Part (d)

Starting with RBI, we can say

$$\begin{aligned} \neg(p \to \neg q) &\equiv \neg(\neg p \vee \neg q) && \text{(RBI)} \\ &\equiv \neg(\neg p) \wedge \neg(\neg q) && \text{(De Morgans Law)} \\ &\equiv p \wedge q && \text{(Double Negation Law)} \end{aligned}$$

Consequently, $p \wedge q$ and $\neg(p \to \neg q)$ are logically equivalent.

Solution - Part (e)

Starting with RBI, we can say

$$\begin{aligned} \neg(p \to q) &\equiv \neg(\neg p \vee q) && \text{(RBI)} \\ &\equiv \neg(\neg p) \wedge \neg q && \text{(De Morgans Law)} \\ &\equiv p \wedge \neg q && \text{(Double Negation Law)} \end{aligned}$$

Consequently, $\neg(p \to q)$ and $p \wedge \neg q$ are logically equivalent.

Insights

Most of these proofs started with RBI, where both De Morgan's Law and the Double Negation Law showed up frequently as well.

Problem 2 Summary

Procedure

(a) Part (a)

- Answer the truth values for each propositional statement and the final propositional statement

(b) Part (b)

- Use RBI, Commutative Property, Double Negation Law to conclude the proof

(c) Part (c)

- Use RBI and Double Negation to reach conclusion

(d) Part (d)

- Use RBI, De Morgans Law, and Double Negation Law to reach conclusion

(e) Part (e)

- Use RBI, De Morgans Law, and Double Negation Law to reach conclusion

Key Concepts

- This problem uses several properties that can be applied to propositional statements and proves that two statements are equivalent to one another
- RBI (Rules By Inference) is $\alpha \to \beta \equiv \neg\alpha \vee \beta$ and is used to take a conditional statement and convert it to a disjunction
- De Morgans Law is $\neg(\alpha \vee \beta) \equiv \neg\alpha \wedge \neg\beta$ and $\neg(\alpha \wedge \beta) \equiv \neg\alpha \vee \neg\beta$ and is used in distributing a negation into a conjunction or disjunction
- Double Negation Law is $\neg(\neg\alpha) \equiv \alpha$ and is used to simplify a doubly negated proposition

Variations

- We could be given a different propositional statement to prove
  - In this case we would use the same properties with the new statement to prove the theorem that has been presented to us

Problem 3

Problem Statement

We can use Table 6 p.27 to prove new ‘rules’. Let’s prove a Logic version of the algebraic version FOIL. You will need to use a compound substitution with an equivalence from Table 6 twice. (optional/recommended - create a truth table as well).

$$(a \wedge b) \vee (c \wedge d) \equiv (a \vee c) \wedge (b \vee c) \wedge (a \vee d) \wedge (b \vee d)$$

Solution

Starting with $(a \wedge b) \vee (c \wedge d)$, we can show

$$\begin{aligned}
(a \wedge b) \vee (c \wedge d) &\equiv ((a \wedge b) \vee c) \wedge ((a \wedge b) \vee d) && \text{(Distributive Laws)}\\
&\equiv (c \vee (a \wedge b)) \wedge (d \wedge (a \wedge b)) && \text{(Commutative Laws)}\\
&\equiv ((c \vee a) \wedge (c \vee b)) \wedge ((d \vee a) \wedge (d \vee b)) && \text{(Distributive Laws)}\\
&\equiv (c \vee a) \wedge (c \vee b) \wedge (d \vee a) \wedge (d \vee b) && \text{(Associative Laws)}\\
&\equiv (a \vee c) \wedge (b \vee c) \wedge (a \vee d) \wedge (b \vee d) && \text{(Commutative Laws)}
\end{aligned}$$

Consequently, $(a \wedge b) \vee (c \wedge d)$ and $(a \vee c) \wedge (b \vee c) \wedge (a \vee d) \wedge (b \vee d)$ are logically equivalent.

$(a \wedge b)$ was treated as a single variable for the first line of this proof. Instead of complicating the above lines with another variable such as α, I chose to wrap it with parenthesis to make the math clearer to me. After the second instance of the distributive law, I wrapped the disjunctions with parenthesis as well to make them resemble a single proposition instead of using a compound substitution.

Problem 3 Summary

Procedure

- Use the Distributive law on the compound statement that is in the premise
- Use the Commutative and Associative Laws along with the Distributive law to reach the conclusion

Key Concepts

- The Distributive law is $\alpha \vee (\beta \vee \gamma) \equiv (\alpha \vee \beta) \vee (\alpha \vee \gamma)$ and is used to distribute propositions in a statement
- The Commutative law is $\alpha \vee \beta \equiv \beta \vee \alpha$ and is used to rearrange propositions in a propositional statement
- The Associative law is used to remove parenthesis in a propositional statement to clean up the presentation of a statement

Variations

- We could be asked to prove a different theorem with the same properties
  - In this case we would use the same properties as before and use them to reach the final conclusion

Problem 4

Problem Statement

Rosen, page 53. Do #8 as warm-up on your own. Do #9 below. Add notes and insight for full credit.

9.) Let $P(x)$ be the statement 'x can speak Russian' and let $Q(x)$ be the statement 'x knows the computer language C++.' Express each of these sentences in terms of $P(x)$, $Q(x)$, quantifiers, and logical connectives. The domain for quantifiers consists of all students at your school.

(a) There is a student at your school who can speak Russian and who knows C++.

(b) There is a student at your school who can speak Russian but who doesn't know C++.

(c) Every student at your school either can speak Russian or knows C++.

(d) No student at your school can speak Russian or knows C++.

Solution - Part (a)

'There is a student at your school who can speak Russian and who knows C++'.

$$\exists x(P(x) \wedge Q(x)) \tag{1}$$

Here we use $\exists$ to denote that over the entire domain of students, we are only concerned that there is at least one student who meets this criteria. We use x to symbolize the domain of all students at the school and we use the conjunction '$\wedge$' to imply that the student knows Russian and C++. If the student knows Russian, $P(x)$ is true and if they know C++ as well, $Q(x)$ is true. The aforementioned statement will only be true if both $P(x)$ and $Q(x)$ applies to the current student.

Solution - Part (b)

'There is a student at your school who can speak Russian but who doesn't know C++.'

$$\exists x(P(x) \wedge \neg Q(x)) \tag{2}$$

Similar to part (a), we once again use the existential quantifier to denote that we are only concerned if one or more students meet this criteria. Here, the only difference from part (a) is that we want to implicate that this supposed student knows Russian but not C++. If $Q(x)$ applies to the student, then the aforementioned statement will be false, just as if $P(x)$ does not apply to the student. The negation operator handles the scenario for when the student knows C++, the statement will also evaluate to false if the student does not know Russian.

Solution - Part (c)

'Every student at your school either can speak Russian or knows C++.'

$$\forall x(P(x) \vee Q(x)) \tag{3}$$

In this example, we use $\forall$ to implicate that for the following statement to be true it must apply to all of the students at the school. After the universal quantifier, we include the domain of students to imply that for the disjunction $P(x) \vee Q(x)$ to be true, every student in the domain must either know Russian or C++. The statement will only evaluate to false for if there is a student that doesn't know Russian or C++.

Solution - Part (d)

'No student at your school can speak Russian or knows C++.'

$$\forall x \neg(P(x) \vee Q(x)) \tag{4}$$

Finally, for this example we once again use the universal quantifier with the entire domain of students. We apply the negation operator '$\neg$' to imply that no student at the school can speak Russian or knows C++. If the inside expression $(P(x) \vee Q(x))$ evaluates to true, then this implies that there is a student that either knows Russian or C++. This in turn would cause the statement to evaluate to false because we are constituting that there is not a single student that this applies to.

Problem 4 - Summary

Procedure

(a) Part (a)

- Use the existential quantifier $\exists$ when dealing with a statement that has the phrase 'There' or something similar in it
- Use a conjunction $\wedge$ when dealing with a statement that has the phrase 'and' in it

(b) Part (b)

- Use the existential quantifier $\exists$ when dealing with a statement that has the phrase 'There' or something similar in it
- Use a conjunction $\wedge$ when dealing with a statement that has the phrase 'but' in it
- Use a negation $\neg$ when dealing with a statement that has the phrase 'doesn't' or something similar in it

(c) Part (c)

- Use the universal quantifier $\forall$ when dealing with a statement that has the phrase 'Every' or something similar in it
- Use a disjunction $\vee$ when dealing with a statement that has the phrase 'or' in it

(d) Part (d)

- Use the universal quantifier $\forall$ when dealing with a statement that has the phrase 'no' or something similar in it
- Use a disjunction $\vee$ when dealing with a statement that has the phrase 'or' in it
- Place the negation outside the propositions when the negation applies to the entire propositional statement

Key Concepts

- Use the existential quantifier $\exists$ when dealing with a statement that is looking for a condition where there is at least one example of this being true
- Use the universal quantifier $\forall$ when dealing with a statement that is looking for a condition where it applies to everything in a statement
- Negations are placed based upon how they are supposed to negate a propositional statement or proposition alone

Variations

- We could be given similar statements and asked to translate them into predicates and quantifiers
  - In this case we would use the same procedure in determining how the and where to place the quantifiers and predicates with the correct logical connectives

Problem 5

Problem Statement

#33 section 1.5, page 67 (add insights and comments for full credit).

33.) Rewrite each of these statements so that negations appear only within predicates (that is, so that no negation is outside a quantifier or an expression involving logical connectives).

(a) $\neg\forall x\forall y P(x, y)$

(b) $\neg\forall y\exists x P(x, y)$

(c) $\neg\forall y\forall x(P(x, y) \vee Q(x, y))$

(d) $\neg(\exists x\exists y\neg P(x, y) \wedge \forall x\forall y Q(x, y))$

(e) $\neg\forall x(\exists y\forall z P(x, y, z) \wedge \exists z\forall y P(x, y, z))$

Solution - Part (a)

Original expression: $\neg\forall x\forall y P(x, y)$.

$$\exists x\exists y\neg P(x, y) \tag{1}$$

These two expressions are logically equivalent because the original is making the claim that it is not the case that for all elements in the domain the predicate is true. This is the same as saying that in the domain there is at least one case where the predicate is false.

Solution - Part (b)

Original expression: $\neg\forall y\exists x P(x, y)$.

$$\exists x\forall y\neg(P(x, y)) \tag{2}$$

These two expressions are logically equivalent because the original is making the claim that it is not the case that for all elements in the domain x and y such that the predicate is true. This is the same as saying that there is at least one element in the domain of x for all elements in y such that the predicate is false.

Solution - Part (c)

Original expression: $\neg\forall y\forall x(P(x, y) \vee Q(x, y))$.

$$\exists y\exists x(\neg P(x, y) \wedge \neg Q(x, y)) \tag{3}$$

The original expression is conveying that it can't be the case that all elements in x and y satisfy the condition where either $P(x, y)$ or $Q(x, y)$ is true. The modified expression states that there is at least an element in x and y where both $P(x, y)$ and $Q(x, y)$ are false.

Solution - Part (d)

Original expression: $\neg(\exists x\exists y\neg P(x, y) \wedge \forall x\forall y Q(x, y))$.

$$(\forall x\forall y P(x, y)) \vee (\exists x\exists y\neg Q(x, y)) \tag{4}$$

The simplest way I can attempt to explain this is that this is an example of how De Morgans Law applies to expressions with universal and existential quantifiers. Here, our conjunction flips to a disjunction and the succeeding predicate $Q(x, y)$ is negated.

Solution - Part (e)

Original expression: $\neg\forall x(\exists y\forall z P(x,y,z) \land \exists z\forall y P(x,y,z))$.

$$\exists x(\forall y\exists z\neg P(x,y,z) \lor \forall z\exists y\neg P(x,y,z)) \tag{5}$$

This is once again another example of De Morgans Law with predicates and quantifiers. I have noticed a pattern here, when a negation operator is carried through an expression, it will flip a existential quantifier to a universal and vice versa. Here our conjunction flips to a disjunction and our predicates are negated via the De Morgans Law.

Problem 5 Summary

Procedure

(a) Part (a)

- Flip the universal quantifiers to existential quantifiers when the negation appears in front of them

(b) Part (b)

- Flip the universal quantifiers to existential quantifiers when the negation appears in front of them
- Flip the existential quantifiers to universal quantifiers when the negation appears in front of them

(c) Part (c)

- Flip the universal quantifiers to existential quantifiers when the negation appears in front of them
- Apply De Morgans Law to the propositional statement with the predicates

(d) Part (d)

- Flip the universal quantifiers to existential quantifiers when the negation appears in front of them
- Flip the existential quantifiers to universal quantifiers when the negation appears in front of them
- Apply De Morgans Law to the propositional statement with the predicates
- Always place the negation of a predicate after the quantifiers in a propositional statement

(e) Part (e)

- Flip the universal quantifiers to existential quantifiers when the negation appears in front of them
- Flip the existential quantifiers to universal quantifiers when the negation appears in front of them
- Apply De Morgans Law to the propositional statement with the predicates
- Always place the negation of a predicate after the quantifiers in a propositional statement

Key Concepts

- Negations flip universal quantifiers to existential quantifiers and vice versa
- De Morgans Law works with propositional statements involving predicates
- Always place the negation of a predicate after the quantifiers in a propositional statement

Variations

- We could be given different statements with negations and asked how to apply them to the new statement
  - We would use the same rules and procedures in determining what a negation does to quantifiers and propositional statements

Problem 6

Problem Statement

Now let's go back to section 1.1. Read the examples on page 11 and 12 to see an important computer science application. Do each of #43 and check your answers in the back of the book. Add notes and insights.

43.) Find the bitwise *OR*, bitwise *AND*, and bitwise *XOR* of each of these pairs of bit strings.

(a) 101 1110, 010 0001

(b) 1111 0000, 1010 1010

(c) 00 0111 0001, 10 0100 1000

(d) 11 1111 1111, 00 0000 0000

Solution - Part (a)

Original: 101 1110, 010 0001.

OR: 111 111 AND: 000 0000 XOR: 111 1111 (1)

Solution - Part (b)

Original: 1111 0000, 1010 1010

OR: 1111 1010 AND: 1010 0000 XOR: 0101 1010 (2)

Solution - Part (c)

Original: 00 0111 0001, 10 0100 1000

OR: 10 0111 1001 AND: 00 0100 0000 XOR: 10 0011 1001 (3)

Solution - Part (d)

Original: 11 1111 1111, 00 0000 0000

OR: 11 1111 1111 AND: 00 0000 0000 XOR: 11 1111 1111 (4)

Solution Insights

This problem seemed to be really easy for me. In the bitwise OR, the corresponding truth value will be 1 if either bits are a 1 and will be 0 if both bits are 0. For the AND, both bits of the bit strings that are being compared must be 1 for it to be 1 and otherwise it will be 0. For the XOR, when comparing the strings, the comparison will only be 1 if one bit from one string is a 1 and the other is a 0. Otherwise, if both are 1's or both are 0's then XOR will evaluate to 0.

Problem 6 Summary

Procedure

- If either bit is a 1 in bitwise OR, then place a 1 for that value, 0 otherwise
- If both bits are 1's in bitwise AND, then place a 1 for that value, 0 otherwise
- If one bit is a 1 and the other is a 0 in bitwise XOR, then place a 1 for that value, 0 otherwise

Key Concepts

- Bitwise OR evaluates to 1 if either bit is a 1
- Bitwise AND evaluates to 1 if both bits are a 1
- Bitwise XOR evaluates to 1 if and only if 1 bit is a 1 and the other is a 0

Variations

- We could be given different strings of bits to evaluate the truth value of each operation
  - We would then use the same rules for each operation to find the truth value of each bit