Page 241

An Introduction to STEM Programming with Python — 2019-09-03a Bonus Chapter 4 — Simplification of Boolean Expressions

Bonus Chapter 4 — Simplification of Boolean Expressions

Introduction

In chapter 5 you were introduced to comparisons and the three basic Boolean operations of AND, OR, and NOT. As you can imagine and have experienced, Boolean expressions may become very complex in a program. This chapter introduces the postulates and theorems used in Boolean algebra and shows how to use them to simplify expressions.

The theorems and postulates are very similar to the ones you have learned in your previous algebra courses, but sometimes they take a few extra moments to understand how they work. Remember, we are dealing with Boolean values and the result of an expression is always a Boolean value.

Objectives

Upon completion of this chapter's exercises, you should be able to:

- Identify the postulates and theorems of Boolean Algebra.
- Use Closure, Identity, Commutative, Distributive, and Closure to simplify Boolean expressions.
- Use the Theorems of Boolean Algebra to re-write expressions.

Prerequisites

This chapter should be introduced after Chapter 6.

The Postulates and Theorems

The five postulates of Closure, Identity, Commutativity, Distributivity, and Compliment are the base rules of Boolean algebra that the other rules are based upon. The Laws of Closure, Identity, and Commutativity are the same, with respect to multiplication and addition, as in your previous experiences with algebra. The Distributive law looks confusing, but is true when using Boolean values. And, the Compliment postulate is defined by the basic operations of OR and AND over the set of Boolean numbers (0 and 1).

			Postulates	n
#	Name	Sum of Products	Product of Sums	

Copyright 2019 — James M. Reneau Ph.D. — <u>http://www.syw2l.org</u> — This work is licensed under a <u>Creative Commons Attribution-ShareAlike 4.0 International License</u>.



An Introduction to STEM Programming with Python — 2019-09-03a	
Bonus Chapter 4 — Simplification of Boolean Expressions	

1	Closure	IF X and Y are elements then X+Y and XY are also elements							
2	Identity	$\mathbf{X} + 0 = \mathbf{X}$	X * 1 = X						
3	Commutative Law	$\mathbf{X} + \mathbf{Y} = \mathbf{Y} + \mathbf{X}$	XY=YX						
4	Distributive Law	X(Y+Z) = XY + XZ	X + YZ = (X + Y)(X + Z)						
5	Compliment Law	X + X' = 1	X * X' = 0						

By using the five postulates we can prove that the following theorems are true. These nine theorems often make simplification of an expression much easier. Once we can write an expression in a form, we can simply re-write it, using the theorem.

	Theorems										
#	Name	Sum of Products	Product of Sums								
1	Null Law	X + 1 = 1	X * 0 = 0								
2	Involution	X'' = X									
3	Idempotency	$\mathbf{X} + \mathbf{X} = \mathbf{X}$	XX = X								
4	Absorption	X + XY = X	$\mathbf{X}(\mathbf{X} + \mathbf{Y}) = \mathbf{X}$								
5	Simplification	$\mathbf{X} + \mathbf{X'Y} = \mathbf{X} + \mathbf{Y}$	X(X'+Y) = XY								
6	Associative Law	X + (Y + Z) = (X + Y) + Z	X(YZ) = (XY)Z								
7	Consensus	XY + X'Z + YZ = XY + X'Z	(X + Y)(X' + Z)(Y + Z) = (X + Y)(X' + Z)								
8	DeMorgan's Law	(X + Y)' = X' * Y'	(XY)' = X' + Y'								
9	Distributing Nots	(X + Y)(X' + Z) = XZ + X'Y (the	term XX' is zero and is dropped)								

This introduction to Boolean Algebra is too short to go into the proofs of all the theorems but as an example here is the proof for one of them.

Proof of Theorem 1(b) — Null Law

X * 0 = X * 0= X * 0 + 0 = X * 0 + X * X' = X (0 + X') = X * X' = 0

+ 0 Identity (p2a)
+ X * X' Compliment (p5b)
+ X') Distributive (p4a)
X' Identity (p2a)
Compliment (p5b)

Theorem 1b may also be shown with a simple truth table.



Х	0	X * 0
0	0	0
1	0	0

Simplification

1 M

10 M

Boolean expressions become complex, easily. When this happens it is often best to use the Distributive Postulate (and others) to either get the function into a form where it is in the form of a "sum-of-products" or a "product-of-sums". Once you can change an expression into either, you should be able to use Consensus, Absorption, Simplification, and the other theorems and postulates to simplify your expression. In "sum-of-products" form a Boolean expression is shown like AB+CD. In this form all parenthesis are gone. The "product-of-sums" format can be represented by an expression like (A+B) (C+D).

For the simplification examples we will convert each expression to "sum-of-products" format while we go through the process.

Example Simplification — One

$$F(X,Y) = (X+Y')(X)(X'+Y') \qquad Distributive
XXX'+XYY'XY'XY'XY'
XX'+XY'+XYY'XY'X'
XX'+XY'+XYY'X'
XX'+XY'+XY'X'
XX'+XY'+XY'X'
XX'+XY'+XY'X'
XX'+XY'+XY'X'
Remove Duplicates using Idempotency
(0 + XY' + 0Y'
XY'
Example Simplification — Two
$$F(A,B,C) = A'(C+B)+B'(A'+C)+(A+B)(B+C)'
A'(C+B)+B'(A'+C)+(A+B)B'C'
A'(C+B)+B'(A'+C)+(A+B)B'C'
A'(C+B)+B'(A'+C)+(A+B)B'C'
A'(C+A'B+B'A'+B'C+A'B)B'C'
A'(C+A'B+B'A'+B'C+A'B)B'C'
A'C+A'B+B'A'+B'C+B'C'A
A'C+A'B+B'A'+B'C+B'C'A
A'B+B'A'+B'C+B'C'A
A'B+B'C+B'C'A
A'B'C+B'C'A
A'B'C+B'C'A
A'B'C+B'C'A
A'B'C+B'C'A
A'B'C+B'C'A
A'B'C+B'C'A
A'B'C+B'C'A
A'B'B'C+B'C'A
A'B'B'A'+B'C+B'C'A
A'B'B'C+B'C'A
A'B'B'A'+B'C+B'C'A
A'B'B'C+B'C'A
A'B'B'C+B'C'A
A'B'B'A'+B'C+B'C'A
A'B'A'+B'C+B'C'A
A'B'B'A'+B'C+B'C'A
A'B'A'+B'C+B'C'A
A'B'A'+B'C+B'C'A
A'B'A'+B'C+B'C'A
A'B'A'+B'C+B'C'A$$$$

Copyright 2019 — James M. Reneau Ph.D. — <u>http://www.syw2l.org</u> — This work is licensed under a <u>Creative Commons Attribution-ShareAlike 4.0 International License</u>.



Example Simplification — Three



Distributive Idempotency Consensus DeMorgan's DeMorgan's Absorption Simplification Absorption Simplification

t this work at Here

DeMorgan's Law

Distributing Nots

Distributive Law

Idempotency

Identity

Involution

•

•

Important Terms

- Absorption
- Associative Law •
- Closure ٠
- Commutative Law •
- Compliment Law ٠
- Consensus

Exercises

1. Show proof for Theorem 1(a): Null Law X + 1 = 1

2. Show proof for Theorem 2: Involution X'' = X

3. Show proof for Theorem 3: Idempotency a) X + X = X

Copyright 2019 — James M. Reneau Ph.D. — http://www.syw2l.org — This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.

- Null Law
- Postulates
- Simplification
- Theorems

An Introduction to STEM Programming with Python — 2019-09-03a Bonus Chapter 4 — Simplification of Boolean Expressions

Page 245

b) XX = X

4. Show proof for Theorem 4: Absorption
a) X + XY = X
b) X(X+Y) = X

5. Show proof for Theorem 5: Simplification
a) X + X'Y = X + Y
b) X(X' + Y) = XY

6. Show proof for Theorem 6: Associative Law
a) X + (Y + Z) = (X + Y) + Z
b) X(YZ) = (XY)Z

7. Show proof for Theorem 7: Consensus
a) XY + X'Z + YZ = XY + X'Z
b) (X + Y)(X' + Z)(Y + Z) = (X + Y)(X' + Z)

8. Show proof for Theorem 8: DeMorgan's Law
a) (X + Y)' = X' * Y'
b) (XY)' = X' + Y'

b) (XY)' = X' + Y' 9. Using Postulate 4(b) write the expression AB+CD as a product of sums.

.

Answers to Selected Exercises

3A.

X+X = X+X X+X = X*1 + X*1 X+X = X(1+1) X+X = X*1X+X = X

Identity Distributive Addition Identity

9. (A+C)(A+D)(B+C)(B+D)

Word Search



Page 246

An Introduction to STEM Programming with Python — 2019-09-03a Bonus Chapter 4 — Simplification of Boolean Expressions

	d	i	s	t	r	i	b	u	t	i	n	g	•	n	0	t	s	ο	u
	С	t	g	d	С	d	r	a	С	1	n	i	z	t	s	а	С	d	t
	1	d	i	s	t	r	i	b	u	t	i	v	е	•	1	a	w	•	e
	0	i	ο	u	а	i	С	ο	n	s	е	n	s	u	s	g	s	е	У
	s	d	d	r	t	h	е	0	r	е	m	s	s	t	i	g	е	i	v
	u	е	е	р	t	b	0	У	z	i	r	t	n	r	t	е	t	s	n
	r	m	m	а	s	s	0	С	i	а	t	i	v	е	•	1	а	w	h
	е	р	ο	v	h	У	n	•	r	t	р	С	u	s	С	У	р	m	n
	а	ο	r	ο	m	t	i	d	е	n	t	i	t	У	i	а	1	а	n
	b	t	g	С	0	m	р	1	i	m	е	n	t	•	1	а	w	е	e
	s	е	а	С	0	m	m	u	t	а	t	i	v	е	•	1	а	w	1
$\Delta \mathbf{E}$	0	n	n	r	u	s	d	р	ο	i	•	j	s	У	1	t	s	i	i
	r	С	'	•	s	i	m	р	1	i	f	i	С	а	t	i	ο	n	t
	р	У	S	а	u	е	а	s	w	•	i	s	v	n	x	r	n	n	С
	t	ο	•	i	q	d	f	n	ο	i	d	р	d	s	n	i	ο	1	t
	i	х	1	а	t	s	i	ο	1	а	m	w	s	а	n	g	i	t	0
	0	t	а	е	g	•	1	•	i	n	v	0	1	u	t	i	ο	n	w
	n	1	w	i	h	0	0	p	ο	s	t	u	1	а	t	е	s	а	u
	е	n	u	1	1	•	1	a	w	b	е	р	р	t	m	a	1	У	a

Absorption, Associative Law, Closure, Commutative Law, Compliment Law, Consensus, DeMorgan's Law, Distributing Nots, Distributive Law, Idempotency, Identity, Involution, Null Law, Postulates, Simplification, Theorems

http://syw2l.org

References

Copyright 2019 — James M. Reneau Ph.D. — <u>http://www.syw2l.org</u> — This work is licensed under a <u>Creative Commons Attribution-ShareAlike 4.0 International License</u>.



Fr