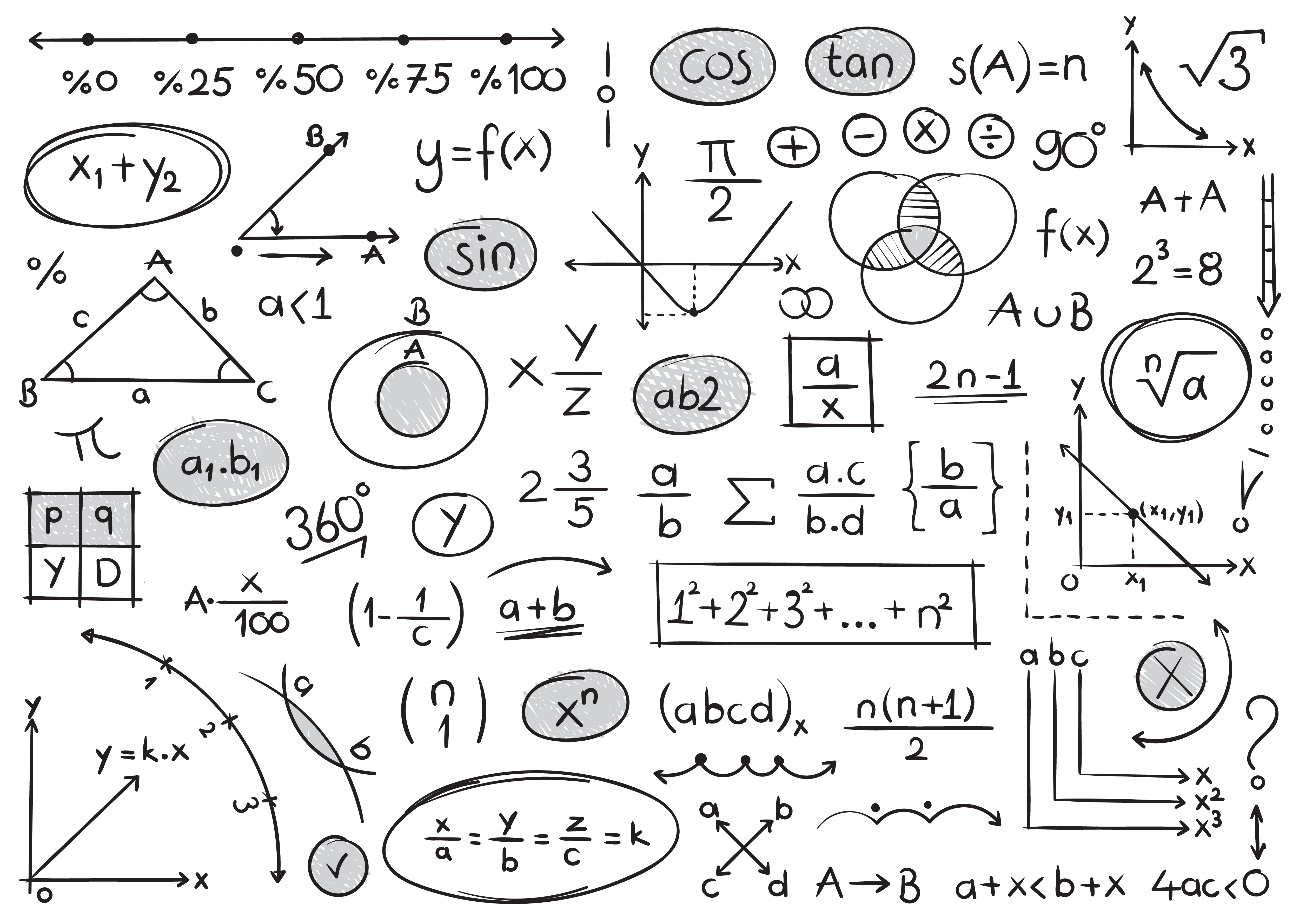
**Introdução da Aula**



**Qual é o foco da aula?**

Nesta aula, trabalharemos determinação e simplificação de expressões lógicas.

**Objetivos gerais de aprendizagem**

Ao longo desta aula, você irá:

* Identificar a relação da porta OR;
* Compreender as regras da álgebra Booleana;
* Interpretar as possibilidades de utilizar até três variáveis nas expressões.

Situação-problema

Vamos continuar nosso aprendizado agora com os estudos sobre expressões lógicas. Para compreender melhor este assunto, é importantíssimo que você aplique nesta aula os conceitos que foram trabalhados na aula 1, na qual tratamos sobre a lógica booleana, portas lógicas e suas respectivas tabela-verdade. Uma sugestão seria rever o conteúdo da aula 1 antes de entrar nesse assunto.

Aqui trabalharemos com determinação e simplificação de expressões lógicas, nas quais toda expressão será resolvida de forma lógica (com resultados 0 e 1). Nestas simplificações usaremos as regras de expressões lógicas e alguns teoremas que nos auxiliam a resolver as expressões lógicas e, a partir disso, chegaremos a um resultado usando o menor número possível de portas lógicas.

Para aplicarmos nosso conhecimento, você deverá criar uma simplificação de uma expressão booleana que será usada em uma placa de circuito, usando as técnicas de álgebra booleana. A expressão a ser simplificada é AB + A(B + C) + B(B + C). Para a simplificação dessa expressão, você deverá usar as regras de álgebra booleana para chegar ao menor número de portas possíveis. Usando o menor número de portas lógicas, diminuímos a quantidade de portas, gerando o mesmo resultado de expressões mais complexas.

Com o objetivo de aprendizagem dessa aula, vamos conhecer e correlacionar a determinação e simplificação de expressões lógicas e, assim, completaremos mais uma fase da nossa competência geral, que é conhecer e compreender os princípios de arquitetura e organização de computadores.

Com foco na simplificação de expressões lógicas, a partir de regras e de teoremas, iremos dedicar-nos a aprender e desenvolver as formas de simplificação para chegarmos ao menor número possível de portas lógicas, fazendo com que usemos o menor número possível de portas lógicas com o mesmo resultado.

**Adição e multiplicação booleana**



Nesta aula iniciaremos os estudos trabalhando com os conceitos envolvidos na matemática dos sistemas digitais, que nada mais é que a álgebra booleana. Para dar andamento a este assunto fica a sugestão de deixar um resumo ao seu lado, para poder consultar os operadores, portas lógicas e suas devidas tabelas-verdade. Elas serão de vital importância para seu desenvolvimento. Desse modo, você já percebeu que o conhecimento básico de álgebra booleana é importantíssimo para este estudo.

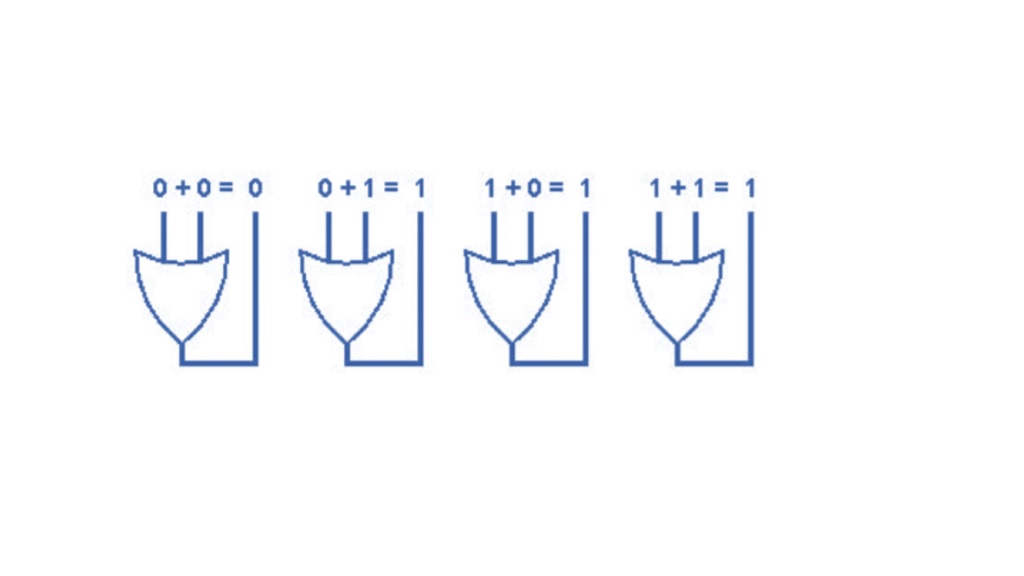
Em álgebra booleana, os termos variável (valor qualquer a ser utilizado), complemento (algo que agregará valor a uma variável) e literal (representação de um valor fixo para uma variável) são utilizados com frequência.

Na aula anterior, usamos o símbolo de “!” (exclamação) para identificar uma negação. Observe como utilizamos este símbolo: Se y = 0 então !y = 1. Aqui, para a negação usaremos o símbolo " "sobre a variável, por exemplo, se A = 0 então Ā= 1.

O símbolo para AND e OR também será invertido, pois na notação matemática usamos: “ ” (ponto) para o AND e “+” (adição) como OR. A maioria absoluta de livros e referências dentro deste assunto usam essas notações. Veja as notações utilizadas no [material](https://wiki.sj.ifsc.edu.br/images/0/0c/CAPITULO2_-_FUNCOES_E_PORTAS_LOGICAS_resumo.pdf) do INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE SANTA CATARINA, CAMPUS SÃO JOSÉ.

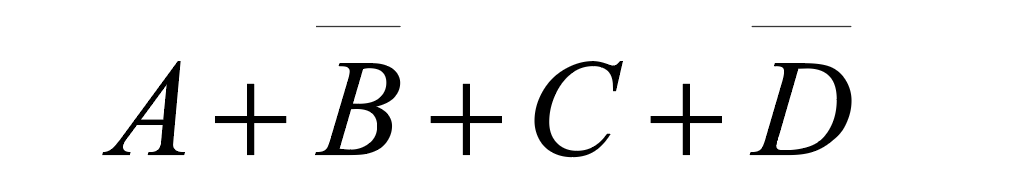
Vamos às operações matemáticas booleanas aplicadas no desenvolvimento de sistemas digitais. Siga em frente!

**Adição booleana** - Quando falamos em adição booleana, estamos falando da porta OR. Regra: Dentro da álgebra booleana, chamamos de **termo-soma**, o que significa que é uma soma de literais (TOCCI, 2011). (Cuidado: O termo se chama termo-soma ou simplesmente adição booleana, porém, o símbolo de + indica a porta lógica OR e não o símbolo de adição que estamos acostumados.)

Ilustração básica com relação a porta OR. Fonte: O autor.

Esta figura representa os dados da tabela-verdade OR, representado anteriormente pela expressão e a seguir pelo seu símbolo lógico.

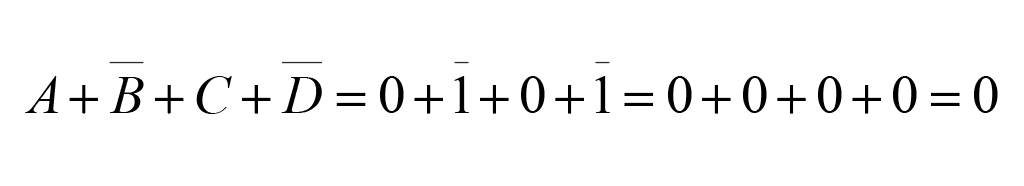
Veja um exemplo: A partir de A, B, C e D, determine os respectivos valores para o termo-soma

seja igual a 0.

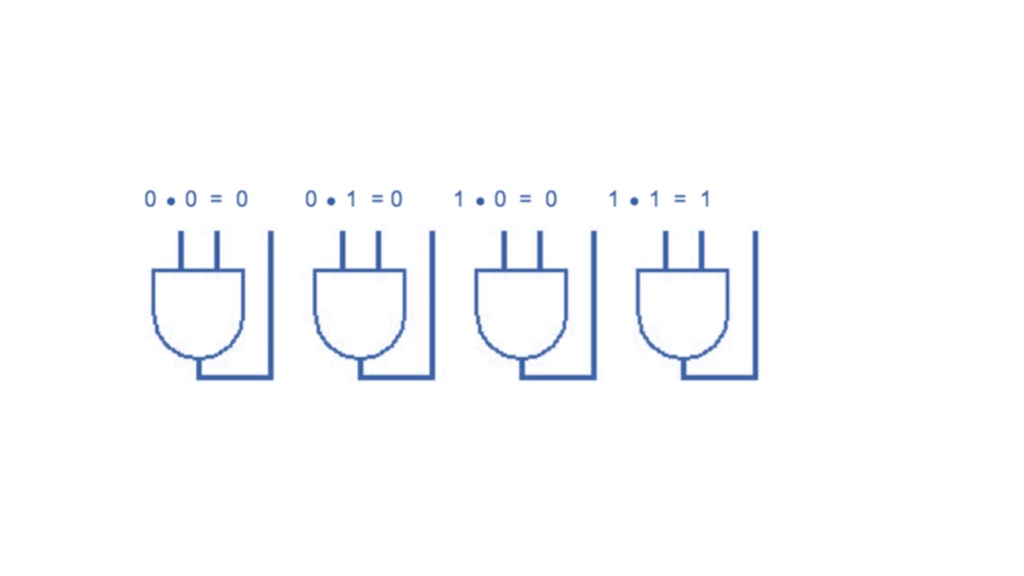
**📝 Exemplificando**

Para que seja 0 o termo-soma, cada uma das literais tem de ser 0. Assim, temos:

A = 0, B = 1, C = 0 e D = 1. B e D possuem a negação, logo o valor deles tem de ser invertido. Veja a expressão toda:



**Multiplicação booleana** - Aqui, quando falamos de multiplicação booleana, estamos falando da porta AND. Regra: Pela álgebra booleana, o **termo-produto** é o produto de literais (aqui, o símbolo • representa o AND e não a multiplicação) (TOCCI, 2011).

Ilustração básica com relação a porta AND. Fonte: O autor.

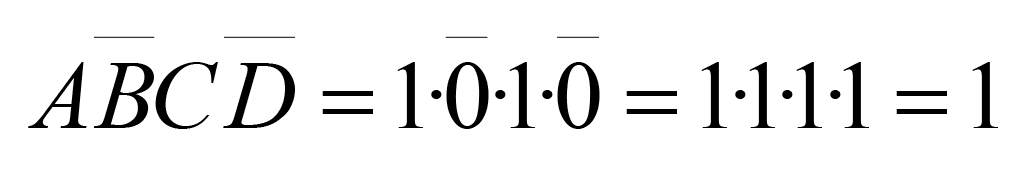
A figura anterior detalha os valores da tabela-verdade AND e, a seguir, a simbologia da porta lógica AND.

Um exemplo da multiplicação: Determine os valores para as literais A, B, C e D que transforme o resultado do termo-produto.

igual a 1.

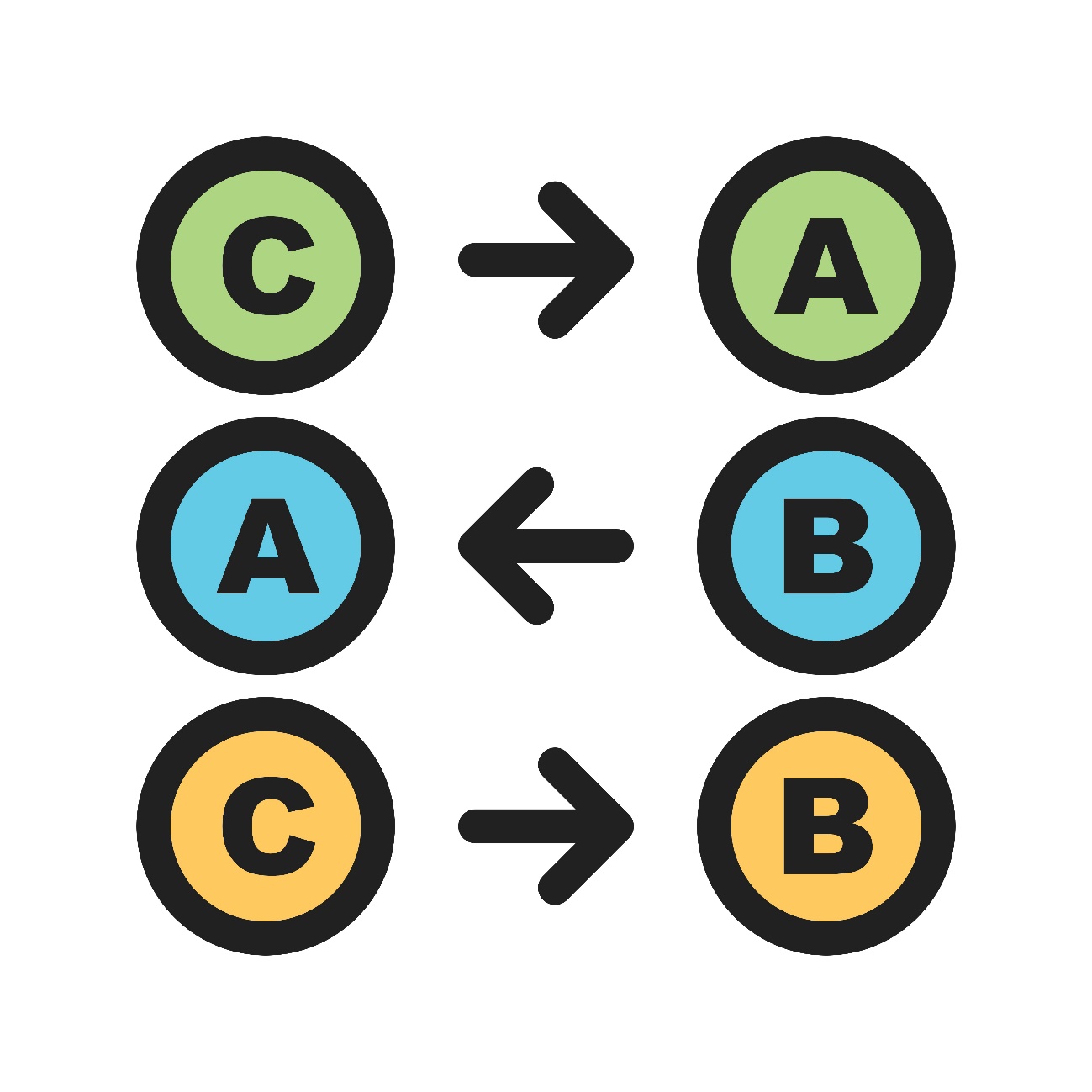
**📝 Exemplificando**

Para o termo-produto ser 1, sabemos pela tabela-verdade AND que somente será verdadeira a expressão que tem dois valores verdadeiros. Aqui é a aplicação. Aplicando no termo-produto temos:



Aqui, podemos aplicar o valor 1 em todas as literais. O resultado obtido foi 1.

**Leis e regras da álgebra booleana e Teoremas de De Morgan**



Quando trabalhamos com as expressões lógicas, temos as leis e regras que regem o uso da álgebra booleana, assim como existem em outras áreas da matemática (SHIMOKAWA, 2014). Essas leis têm de ser seguidas para que a álgebra booleana seja aplicada adequadamente. São elas:

* Leis comutativas da adição e multiplicação.
* Leis associativas da adição e multiplicação.
* Lei distributiva.
* 12 regras básicas da álgebra de Boole.

As leis comutativas da adição e multiplicação, as leis associativas da adição e multiplicação e a lei distributiva são as mesmas leis aplicadas à álgebra comum, que com certeza você já aprendeu no primeiro grau. Vamos revê-las rapidamente:

1. Lei Comutativa da Adição → A + B = B + A.
2. Lei Comutativa da Multiplicação Symbol → AB = BA.
3. Lei Associativa da Adição Symbol → A + (B + C) = (A + B) + C.
4. Lei Associativa da Multiplicação → A(BC) = (AB)C.
5. Lei Distributiva → A(B + C) = AB + AC.

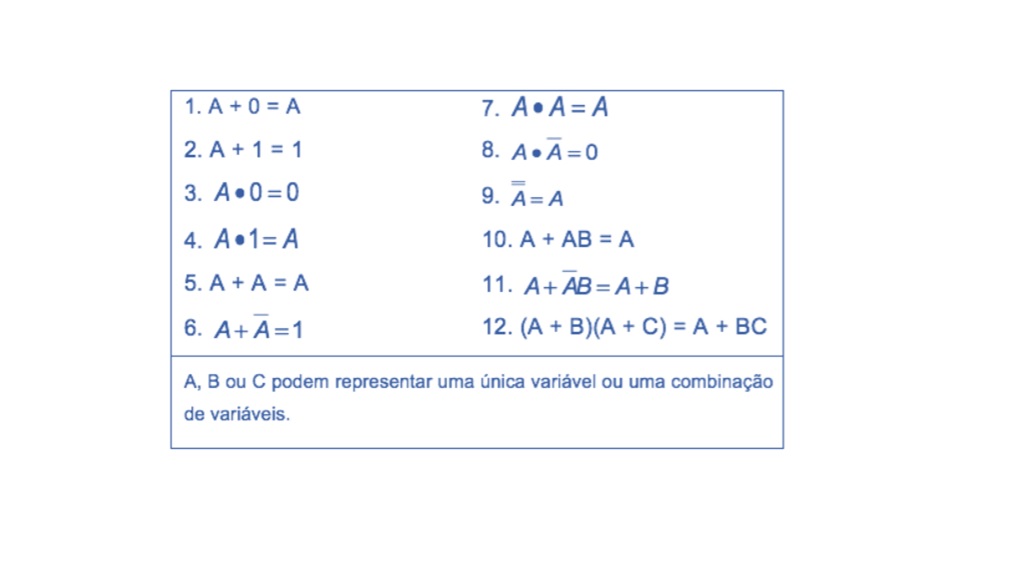
\_\_\_\_\_\_\_

**💭 Reflita**

As leis associativas, comutativas e de distribuição são as mesmas regras utilizadas na álgebra comum da nossa matemática (TOCCI; WIDMER, 2011).

\_\_\_\_\_\_\_

E agora partimos para as regras da álgebra booleana. São elas:

Regras da Álgebra Booleana. Fonte: Adaptado de Shimokawa (2014).

Se você analisar logicamente cada uma delas, verá que são todas fáceis, pois conseguiu compreender cada expressão lógica (e a maioria usa a lógica das tabelas verdade já estudadas). Veja alguns exemplos:

1. Regra 1 → A + 0 = A → Para qualquer valor de A (0 ou 1) com a porta lógica OR, dará sempre o resultado do valor de A (lembra da tabela verdade?).
2. Regra 5 → A + A = A → Para qualquer valor de A (0 ou 1) dará sempre A. Note que se A = 0, teremos 0 + 0 = 0 e se A = 1, teremos 1 + 1 = 1 (também da tabela verdade OR).
3. Regra 9 → A = A → Se a variável A = 0, a primeira negação transforma em A = 1 e a segunda negação transforma em A = 0, portanto, o primeiro valor de A, logo, quando tivermos dupla negação, é a mesma coisa que não ter negação nenhuma.
4. Regra 12 → (A + B)(A + C) = A + BC → Nesta colocamos o A em evidência, ficando A + BC.

**Teoremas de De Morgan** - Dois importantes teoremas de álgebra booleana são uma contribuição de um importante matemático, amigo de Boole, chamado de De Morgan (TOCCI, 2011). Os seus teoremas são:

1. (x̅ + y̅) = x̅ • y̅ → Este teorema diz que, quando a soma lógica (OR) de duas variáveis é invertida, é o mesmo que inverter cada variável individualmente, fazendo a operação AND estre estas variáveis invertidas (TOCCI, 2011).
2. (x̅ • y̅) = x̅ + y̅ → Este teorema diz que, quando o produto (AND) de duas variáveis é invertida, é o mesmo que inverter cada variável individualmente, fazendo a operação OR entre elas.

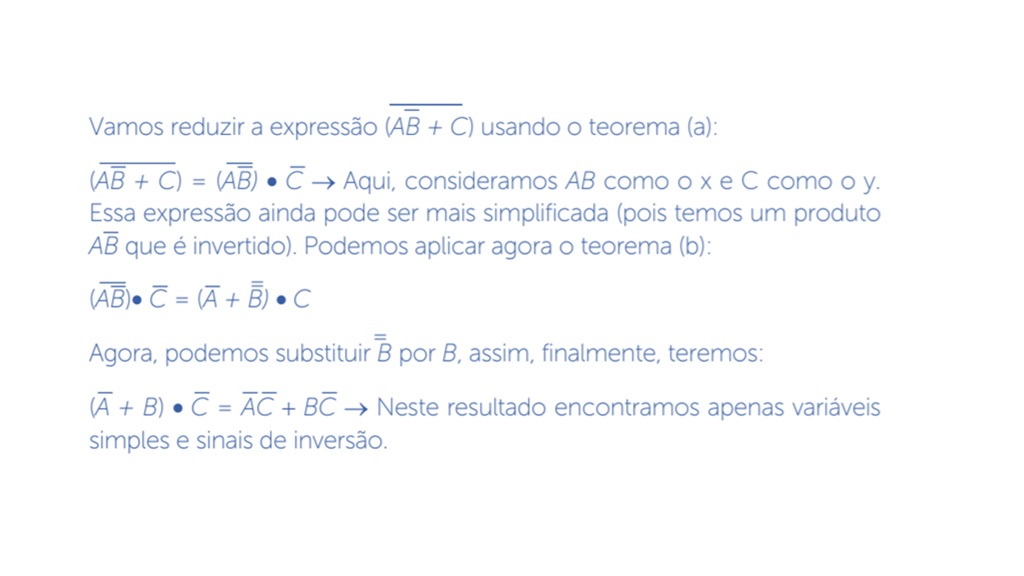
\_\_\_\_\_\_\_

**💭 Reflita**

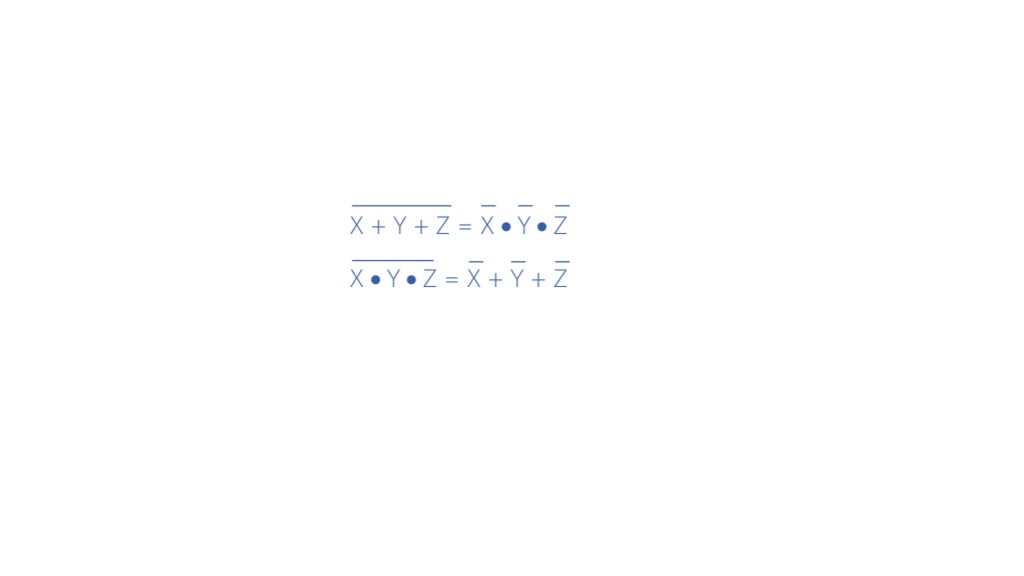
Lembre-se de que as variáveis x e y, utilizadas nas expressões ou teoremas, representam um valor único ou uma expressão que contém mais de uma variável.

\_\_\_\_\_\_\_

**📝 Exemplificando**

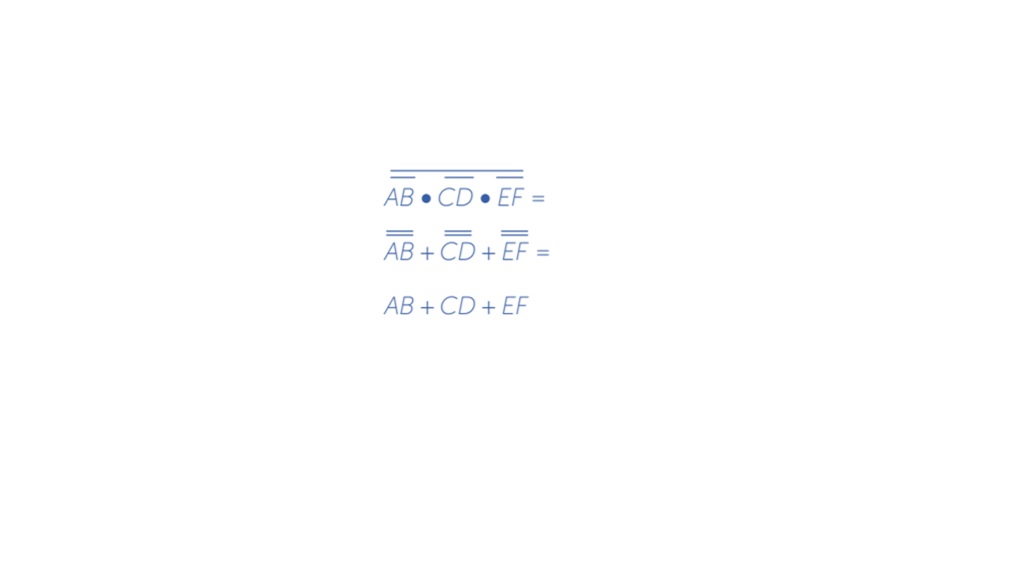


Da mesma maneira que trabalhamos com duas variáveis, podemos trabalhar com três:



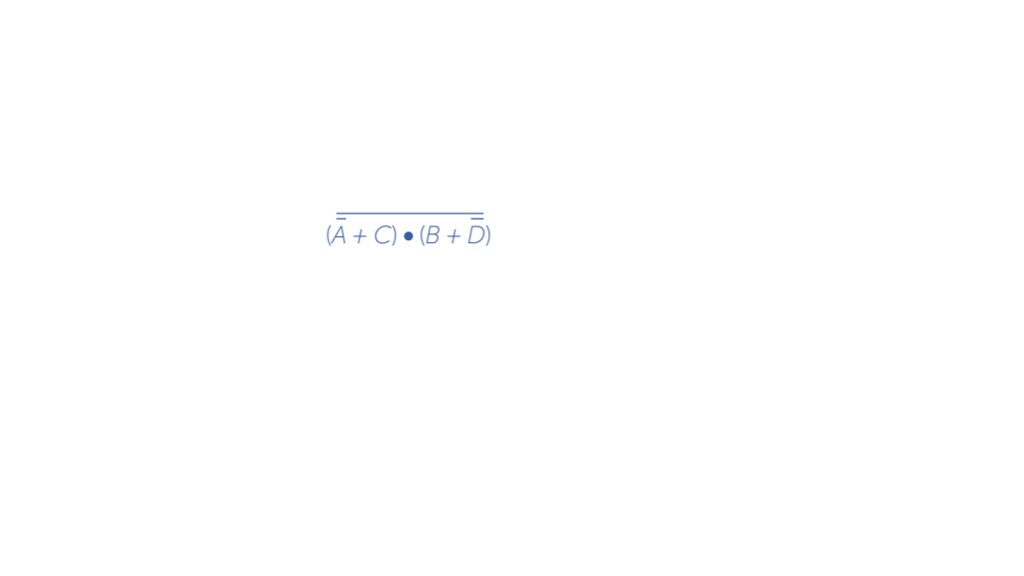
Note o uso de três termos nas expressões anteriores.

Vamos a mais um exemplo de simplificação de expressão, agora usando três variáveis:



**💪 Faça você mesmo**

Simplifique a expressão:



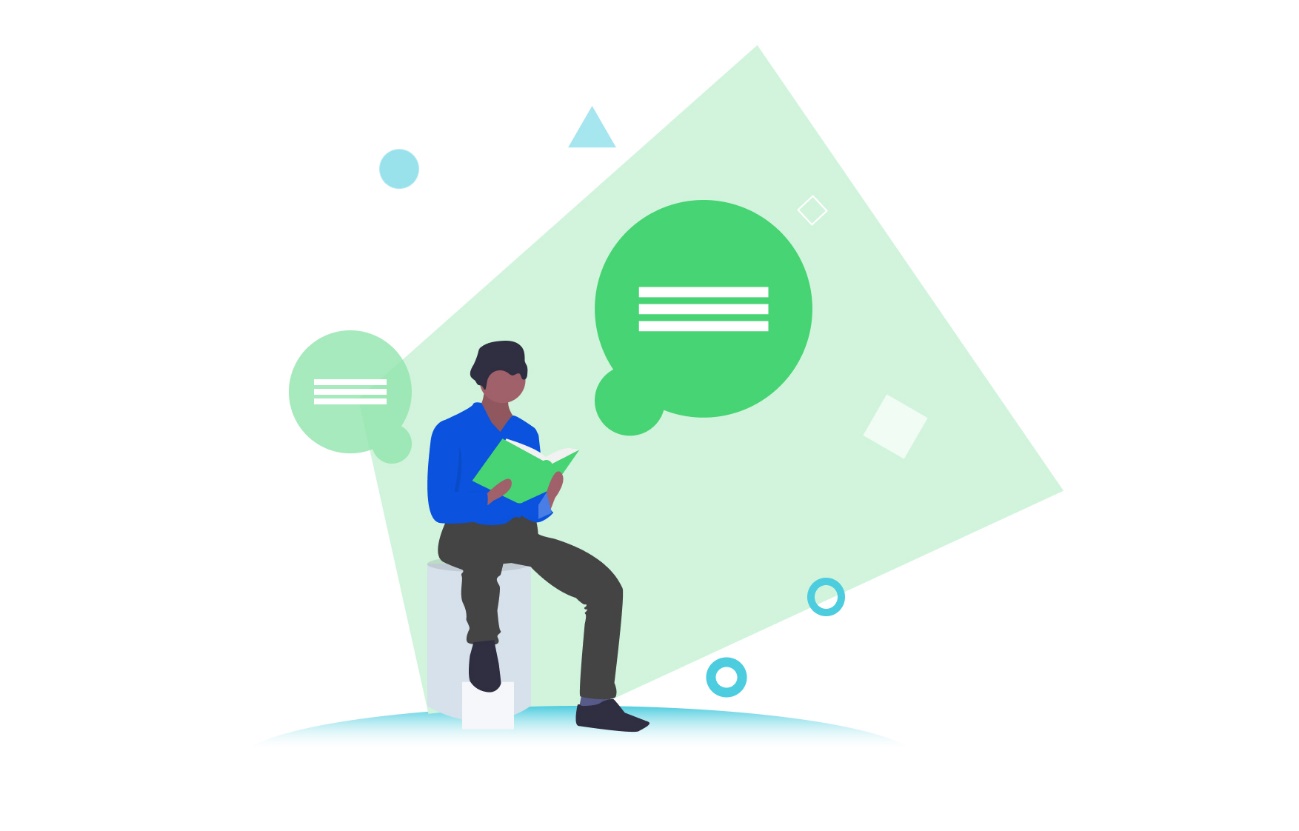
para que tenha somente variáveis simples invertidas (TOCCI, 2011).

\_\_\_\_\_\_\_

**➕ Pesquise mais**

MUITO IMPORTANTE: Veja o vídeo sobre as explicações das Leis e Regras de Álgebra Booleana no [vídeo](https://www.youtube.com/watch?v=wK2YdpY6KXU&feature=youtu.be)“Simplificações de Expressões Booleanas #1: Entendendo”  Assista ao vídeo todo, se possível, senão, assista até os 17m e 33s.

**Conclusão**



Você deverá criar uma simplificação de uma expressão booleana que será usada em uma placa de circuito, usando as técnicas de álgebra booleana. A expressão a ser simplificada é AB + A(B + C) + B(B + C). Para esta simplificação você deverá usar as regras de álgebra booleana e chegar ao menor número de portas possíveis.

Vale lembrar que, quando lemos AB, é o mesmo que A • B.

Para isso vamos usar as regras de Boole, e para um melhor entendimento faremos a simplificação passo a passo:

Expressão a ser simplificada: AB + A(B + C) + B(B + C).

1. Aplicar a lei distributiva no segundo e terceiro termo:

AB + AB + AC + BB + BC =

1. Aplicar a regra 5 (AB + AB = AB) nos dois primeiros termos da expressão:

AB + AC + BB + BC =

1. Agora, aplique a regra 7 (BB = B) no terceiro termo:

AB + AC + B + BC =

Aplicar a regra 10 (B + BC = B) no terceiro e quarto termo:

AB + AC + B

Para finalizar aplicamos a regra 10 (AB + B = B) novamente, agora no primeiro termo e no último termo:

B + AC

Desse modo, não conseguimos mais simplificar a expressão, chegando à sua total simplificação.

A expressão simplificada B + AC representa o menor número de portas possíveis para a expressão lógica B + A(B + C) + B(B + C).

\_\_\_\_\_\_\_

**📌 Lembre-se**

A, B e C são variáveis e podem possuir um valor único ou uma expressão.