

## Algoritmo de Booth

Criado por Andrew D. Booth em 1951, algoritmo de Booth é um algoritmo de multiplicação para números binários com sinal na notação complemento de dois. Booth usava calculadoras de mesa que eram mais rápidas em deslocar do que em somar e criou o algoritmo para aumentar sua velocidade.

### Funcionamento e Exemplo:

x = número de bits da representação binária em complemento de dois do multiplicando.

y = número de bits do multiplicador.

#### Preenchendo a coluna “x”:

Adição: valor do multiplicando em binário.

Subtração: O negativo do multiplicando em binário.

Produto: Zeros.

#### Preenchendo a coluna “y”:

Adição: Zeros.

Subtração: Zeros.

Produto: O multiplicador.

#### Última coluna:

Coloque zero.

ex.: Encontre  $3 \times (-4)$

	X	Y	1
Adição	0011	0000	0
Subtração	1011	0000	0
Produto	0000	11000	0

### Passos para resolução:

Repita o procedimento abaixo de acordo com o número de bits de y:

1. Se os dois últimos bits do produto são:

00 ou 11: não faça nada.

01: Produto = Produto + Adição (ignore o overflow).

10: Produto = Produto + Subtração (ignore o overflow).

2. Desloque o Produto para direita um bit. O sinal do Produto deve ser preservado, ou seja, se o bit mais significativo for 1, então após o deslocamento o novo bit mais significativo também deve ser 1, caso o bit mais significativo for 0, após o deslocamento o novo bit mais significativo também deve ser 0.

3. Descarte o primeiro bit do produto (direita para esquerda) para o resultado final.

No caso de  $3 \times (-4)$ , o loop será executado quatro vezes.

1. Produto = 0000 1100 0. Os últimos 2 bits são 00. Produto = 0000 0110 0. Um deslocamento a direita.
2. Produto = 0000 0110 0. Os últimos 2 bits são 00. Produto = 0000 0011 0. Um deslocamento a direita.
3. Produto = 0000 0011 0. Os últimos 2 bits são 10.  $1101\ 0011\ 0 = \text{Produto} + \text{Subtração}$ . Produto = 1110 1001 1. Um deslocamento a direita.
4. Produto = 1110 1001 1. Os últimos 2 bits são 11. Produto = 1111 0100 1. Um deslocamento a direita. Produto Final = 1111 0100 1, que representa -12.