

Trabalho Prático 3º Bimestre

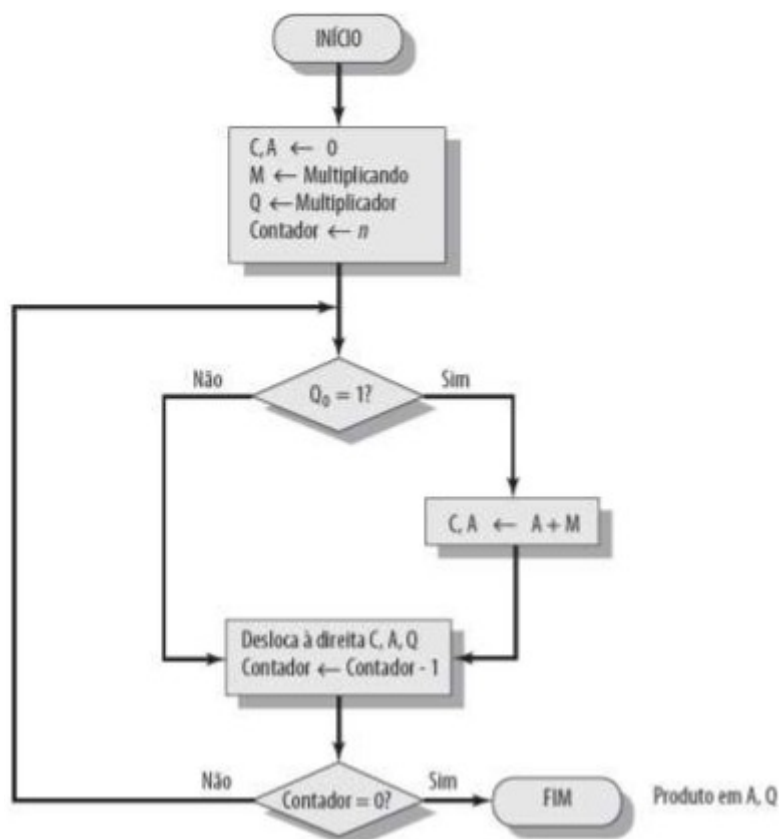
Objetivo:

Aplicar os conhecimentos adquiridos durante o Bimestre para implementar um programa que faça a multiplicação de números binários na representação sinal magnitude.

Descrição:

O trabalho consiste em programar uma **calculadora** que realize uma **multiplicação** de dois números de no máximo **15 bits inteiros e positivos**.

A **calculadora** receberá **dois** números em **decimal**, os quais devem ser convertidos para a representação em binário, e após isso, são realizadas as operações detalhadas no fluxograma abaixo:



Ao fim da execução a calculadora apresentará o resultado final, bem como os valores das variáveis **C, A, M e Q** em **todas** as etapas.

Exemplo:

```
=====
TRABALHO MULTIPLICACAO DE NUMEROS BINARIOS
Professor Guilherme Henrique de Souza Nakahata

UNESPAR - Apucarana

=====
Informe o Multiplicando:
253
Informe o Multiplicador:
9752
Multiplicando em Binario: [1, 1, 1, 1, 1, 1, 0, 1]
Multiplicador em Binario: [1, 0, 0, 1, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0]
=====
Início Iteração
Contador: 15
Q0: 0
C: [0]
A: [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
Q: [0, 1, 0, 0, 1, 1, 0, 0, 0, 0, 1, 1, 0, 0]
M: [0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 0]
Após deslocamento
C: [0]
A: [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
Q: [0, 0, 1, 0, 0, 1, 1, 0, 0, 0, 0, 1, 1, 0]
=====
Início Iteração
Contador: 14
Q0: 0
C: [0]
A: [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
Q: [0, 0, 1, 0, 0, 1, 1, 0, 0, 0, 0, 1, 1, 0]
M: [0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 0]
Após deslocamento
C: [0]
A: [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
Q: [0, 0, 0, 1, 0, 0, 1, 1, 0, 0, 0, 0, 1, 0]
=====
Início Iteração
Contador: 13
Q0: 0
C: [0]
A: [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
Q: [0, 0, 0, 1, 0, 0, 1, 1, 0, 0, 0, 0, 1, 0]
M: [0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 0]
Após deslocamento
C: [0]
A: [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
Q: [0, 0, 0, 0, 1, 0, 0, 1, 1, 0, 0, 0, 0, 1]
=====
```

```

=====
Início Iteração
Contador: 12
Q0: 1
C: [0]
A: [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
Q: [0, 0, 0, 0, 1, 0, 0, 1, 1, 0, 0, 0, 0, 1, 1]
M: [0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 0, 1]
Após a Soma
C: [0]
A: [0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 0, 1]
Após deslocamento
C: [0]
A: [0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 0]
Q: [1, 0, 0, 0, 0, 1, 0, 0, 1, 1, 0, 0, 0, 0, 1]
=====
Início Iteração
Contador: 11
Q0: 1
C: [0]
A: [0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 0]
Q: [1, 0, 0, 0, 0, 1, 0, 0, 1, 1, 0, 0, 0, 0, 1]
M: [0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 0, 1]
Após a Soma
C: [0]
A: [0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 1, 1, 0, 1, 1]
Após deslocamento
C: [0]
A: [0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 1, 1, 0, 1]
Q: [1, 1, 0, 0, 0, 0, 1, 0, 0, 1, 1, 0, 0, 0, 0]
=====

```

```

=====
Início Iteração
Contador: 10
Q0: 0
C: [0]
A: [0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 1, 1, 0, 1]
Q: [1, 1, 0, 0, 0, 0, 1, 0, 0, 1, 1, 0, 0, 0, 0]
M: [0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 0, 1]
Após deslocamento
C: [0]
A: [0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 1, 1, 0]
Q: [1, 1, 1, 0, 0, 0, 0, 1, 0, 0, 1, 1, 0, 0, 0]
=====
Início Iteração
Contador: 9
Q0: 0
C: [0]
A: [0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 1, 1, 0]
Q: [1, 1, 1, 0, 0, 0, 0, 1, 0, 0, 1, 1, 0, 0, 0]
M: [0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 0, 1]
Após deslocamento
C: [0]
A: [0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 1, 1]
Q: [0, 1, 1, 1, 0, 0, 0, 0, 1, 0, 0, 1, 1, 0, 0]
=====

```

```

=====
Início Iteração
Contador: 8
Q0: 0
C: [0]
A: [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 1, 1]
Q: [0, 1, 1, 1, 0, 0, 0, 0, 1, 0, 0, 1, 1, 0, 0]
M: [0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 0, 1]
Após deslocamento
C: [0]
A: [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 1]
Q: [1, 0, 1, 1, 1, 0, 0, 0, 0, 1, 0, 0, 1, 1, 0]
=====

```

```

=====
Início Iteração
Contador: 7
Q0: 0
C: [0]
A: [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 1]
Q: [1, 0, 1, 1, 1, 0, 0, 0, 0, 1, 0, 0, 1, 1, 0]
M: [0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 0, 1]
Após deslocamento
C: [0]
A: [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1]
Q: [1, 1, 0, 1, 1, 1, 0, 0, 0, 0, 1, 0, 0, 1, 1]
=====

```

```

=====
Início Iteração
Contador: 6
Q0: 1
C: [0]
A: [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1]
Q: [1, 1, 0, 1, 1, 1, 0, 0, 0, 0, 1, 0, 0, 1, 1]
M: [0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 0, 1]
Após a Soma
C: [0]
A: [0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0]
Após deslocamento
C: [0]
A: [0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0]
Q: [0, 1, 1, 0, 1, 1, 1, 0, 0, 0, 0, 1, 0, 0, 1]
=====

```

```

=====
Início Iteração
Contador: 5
Q0: 1
C: [0]
A: [0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0]
Q: [0, 1, 1, 0, 1, 1, 1, 0, 0, 0, 0, 1, 0, 0, 1]
M: [0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 0, 1]
Após a Soma
C: [0]
A: [0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 1]
Após deslocamento
C: [0]
A: [0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0]
Q: [1, 0, 1, 1, 0, 1, 1, 1, 0, 0, 0, 0, 1, 0, 0]
=====

```

```

=====
Início Iteração
Contador: 4
Q0: 0
C: [0]
A: [0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0]
Q: [1, 0, 1, 1, 0, 1, 1, 1, 0, 0, 0, 0, 1, 0, 0]
M: [0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 0, 1]
Após deslocamento
C: [0]
A: [0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0]
Q: [0, 1, 0, 1, 1, 0, 1, 1, 1, 0, 0, 0, 0, 1, 0]
=====

```

```

=====
Início Iteração
Contador: 3
Q0: 0
C: [0]
A: [0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0]
Q: [0, 1, 0, 1, 1, 0, 1, 1, 1, 0, 0, 0, 0, 1, 0]
M: [0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 0, 1]
Após deslocamento
C: [0]
A: [0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0]
Q: [0, 0, 1, 0, 1, 1, 0, 1, 1, 1, 0, 0, 0, 0, 1]
=====

```

```

=====
Início Iteração
Contador: 2
Q0: 1
C: [0]
A: [0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0]
Q: [0, 0, 1, 0, 1, 1, 0, 1, 1, 1, 0, 0, 0, 0, 1]
M: [0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 0, 1]
Após a Soma
C: [0]
A: [0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 1, 1, 0, 1]
Após deslocamento
C: [0]
A: [0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 1, 1, 0]
Q: [1, 0, 0, 1, 0, 1, 1, 0, 1, 1, 1, 0, 0, 0, 0]
=====

```

```

=====
Início Iteração
Contador: 1
Q0: 0
C: [0]
A: [0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 1, 1, 0]
Q: [1, 0, 0, 1, 0, 1, 1, 0, 1, 1, 1, 0, 0, 0, 0]
M: [0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 0, 1]
Após deslocamento
C: [0]
A: [0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 1, 1]
Q: [0, 1, 0, 0, 1, 0, 1, 1, 0, 1, 1, 1, 0, 0, 0]
=====

```

```
=====
Final da operação!
Q0: 0
A: [0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 1, 1]
Q: [0, 1, 0, 0, 1, 0, 1, 1, 0, 1, 1, 1, 0, 0, 0]
M: [0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 0, 1]
Resultado em Binário: 000000001001011010010110111000
Resultado em Decimal: 2467256
=====
```

AVALIAÇÃO

A avaliação do trabalho será a soma das seguintes notas:

- Código fonte e executável (0 a 6)
- Apresentação para o professor (0 a 4)

ENTREGA

Cada **aluno** deve entregar **dois** arquivos, um **arquivo fonte do programa** e uma **documentação** com as **instruções** para **compilar** e **rodar** o programa.

O código fonte deve estar comentado.

Trabalhos copiados serão zerados.

A não apresentação para o professor fará com que o trabalho seja zerado.

DATA DE ENTREGA

Envio dos arquivos via e-mail até dia **13/11/2023** para **guilherme.henrique@ies.unespar.edu.br**, ou e-mail com o link para download via Github, Google Drive, Dropbox, etc.

O assunto do e-mail **deve** ser obrigatoriamente “**Trabalho de Arquitetura – 3º Bimestre – C.C UNESPAR**”

Data da apresentação: **14/11/2023**