

## Trabalho de Programação – Processador INTEL – 2025/2

### Descrição Geral

Desenvolva um programa na linguagem de montagem do INTEL X86 capaz de ler um arquivo de entrada que contém uma matriz de números inteiros, e aplicar sobre esta matriz uma sequência de operações determinadas pelo operador do programa.

Os números da matriz poderão ser representados internamente no programa por valores inteiros de 16 bits com sinal.

### Formato do arquivo de entrada

O arquivo com a matriz a ser lida terá, sempre, o nome “MAT.TXT”.

Os números no arquivo de entrada estão organizados em linhas e colunas. São “N” linhas e “N+1” colunas. O número de linhas “N” pode assumir qualquer valor entre 2 e 7.

Os números de uma linha estão separados por um caractere “;”. É garantido que não existam espaços de qualquer tipo (SPACE, TAB, etc) entre os caracteres que formam a linha. Estes números podem ser positivos (sem qualquer caractere de indicação do sinal) ou negativos (com o caractere “-” indicando o sinal).

É garantido que não existam linhas vazias, exceto, possivelmente, as últimas linhas do arquivo. Portanto, para determinar o valor de “N”, o programa deve verificar a quantidade de colunas (números separados por “;”) existentes nas linhas.

O programa deve verificar se o arquivo de entrada obedece às regras definidas anteriormente. Caso isso não ocorra, o programa deve emitir uma mensagem informando o erro encontrado e, se possível, a linha em que foi identificado o erro.

Exemplo de arquivo MAT.TXT

```
4;-2;1;3  
2;1;-1;1  
1;-1;3;8
```

Ao ler este arquivo o programa deve identificar que “N=3”.

### Operação do programa

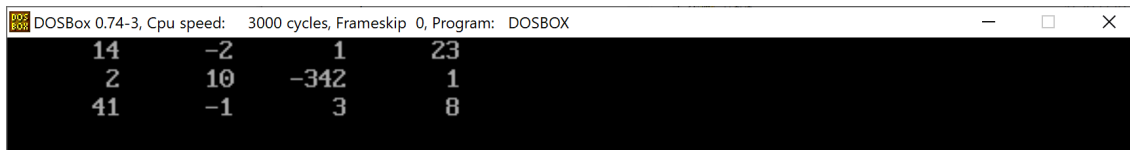
O programa deverá operar da seguinte forma:

- (1) Ler a matriz de números do arquivo de entrada.
- (2) Apresentar a matriz na tela (ver “Apresentação da matriz na tela”).
- (3) Aguardar por um comando do operador e executá-lo. São quatro comandos possíveis: MUL, ADD, UNDO e WRITE.
- (4) Após a execução dos comandos MUL, ADD e UNDO, retornar para a etapa (2). Após a execução do comando WRITE, retornar para a etapa (3).

### Apresentação da matriz na tela

A matriz deve ser apresentada na tela, linha por linha, com as colunas separadas de maneira que os números estejam alinhados à direita. Cada coluna deve ocupar 8 espaços (dois espaços em branco e seis espaços para o número e um eventual caractere “-”).

Na figura abaixo pode-se ver um exemplo deste formato.



The screenshot shows a DOSBox window with the title bar 'DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX'. The window displays a 4x4 matrix of integers on a black background with white text. The matrix is as follows:

14	-2	1	23
2	10	-342	1
41	-1	3	8

## Comandos

---

O programa deve ser capaz de realizar quatro comandos, que serão descritos a seguir.

### 1. Comando MUL

Este comando indica ao programa para multiplicar todos os elementos de uma LINHA da matriz por uma CONSTANCE. O comando deve ser acompanhado pelos seguintes dois parâmetros:

- LINHA, que deve estar entre 1 e N, e informa o número de uma linha da matriz;
- CONSTANCE, valor de 16 bits com sinal em complemento de 2.

### 2. Comando ADD

Este comando indica ao programa para somar todos os elementos da LINHA\_ORG com a LINHA\_DST e escrever o resultado da soma na LINHA\_DST. O comando deve estar acompanhado pelos seguintes dois parâmetros:

- LINHA\_DST, que deve estar entre 1 e N, e informa o número da linha da matriz que receberá o resultado das somas;
- LINHA\_ORG, que deve estar entre 1 e N, e informa o número da linha da matriz a ser somada com a LINHA\_DST.

### 3. Comando UNDO

Este comando indica ao programa para desfazer a última operação efetuada. O programa deverá ser capaz de desfazer apenas a última operação efetuada. Este comando não requer parâmetros.

### 4. Comando WRITE

Este comando indica ao programa para gravar a matriz em um arquivo de saída. O comando deve estar acompanhado pelo seguinte parâmetro:

- NOME, que indica o nome do arquivo onde gravar a matriz.

A matriz deve ser acrescida ao arquivo. Ou seja, o conteúdo original do arquivo deve ser mantido e a matriz deverá ser escrita ao seu final.

A matriz deve ser escrita no arquivo de saída usando o mesmo formato do arquivo de entrada. Ou seja, "N" linhas com "N+1" colunas de números separados por ";;".

## Correção dos Trabalhos

---

Para corrigir os trabalhos serão utilizados arquivos de entrada mal formados (para identificar a capacidade do programa em verificar erros nos arquivos de entrada) e bem formados.

Os arquivos fonte entregues serão montados usando o MASM no ambiente DOSBOX. Programas que possuam erros de montagem receberão nota 0 (zero).

O código do programa fonte deverá conter comentários descritivos da implementação.

O trabalho deverá ser entregue até a data prevista indicada no sistema Moodle. Não serão aceitos trabalhos entregues após o prazo estabelecido. Os trabalhos não entregues até a data prevista receberão nota zero.

### **O que deve ser entregue?**

---

Deverá ser entregue somente o arquivo fonte (arquivo ASM) escrito na linguagem simbólica do INTEL X86, com a solução do problema apresentado, no Moodle da disciplina. **Não entregue o arquivo EXE.**

O programa fonte deverá conter comentários descritivos da implementação. Por exemplo, nos comentários podem ser usados comandos da linguagem "C".

O trabalho deverá ser entregue até a data especificada no link de entrega no sistema Moodle. **Não serão aceitos trabalhos após o prazo estabelecido.**

### **Observações**

---

Recomenda-se a troca de ideias entre os alunos. Entretanto, a identificação de cópias de trabalhos acarretará na aplicação do Código Disciplinar Discente e a tomada das medidas cabíveis para essa situação. Inicialmente, nesses casos, **ambos os trabalhos: original e cópias, receberão nota zero.**

O professor da disciplina reserva-se o direito, caso necessário, de solicitar uma demonstração do programa, onde o aluno será arguido sobre o trabalho como um todo. Nesse caso, a nota final do trabalho levará em consideração o resultado da demonstração.