

O objetivo desta atividade é permitir que o aluno seja capaz de criar um programa em linguagem assembly que utilize comandos de repetição.

Parte 1 – Comando For em assembly

A implementação de comandos de repetição em linguagem assembly, como o comando For da linguagem C, é sempre feita combinando o uso de contadores (que podem ser registradores ou variáveis em memória) e instruções de saltos condicionais.

O exemplo a seguir ilustra como implementar um programa em assembly que exibe o caracter “*” na tela 50 vezes.

Ex:

```
TITLE For
.MODEL SMALL
.STACK 100h
.CODE
; Inicializa o contador (neste caso o registrador BL) com o valor 50
MOV BL,50
; Define um rótulo chamado INICIO
INICIO:
; Exibe o caracter “*” na tela
MOV AH,2
MOV DL, “*”
INT 21h
; Decrementa o contador (BL)
DEC BL
; Caso o contador (BL) seja diferente de zero: salta para INICIO
; Caso o contador (BL) seja igual a zero: continua a execução sequencialmente
JNZ INICIO
; Finaliza o programa
MOV AH,4Ch
INT 21h
END
```

Parte 2 – Instrução LOOP

A instrução **LOOP** pode ser utilizada para a implementação de comandos de repetição, como o comando For mostrado no exemplo anterior.

A instrução **LOOP** sempre utiliza como contador o **registrador CX** e:

- salta para o **ROTULO** especificado na instrução caso o **registrador CX** seja diferente de zero; ou
- continua a execução sequencialmente caso o **registrador CX** seja igual a zero.

Podemos dizer dessa forma que a instrução:

LOOP ROTULO

É equivalente às instruções:

DEC CX

JNZ ROTULO

Poderíamos, por exemplo, reescrever o programa anterior, utilizando:

1. o **registrador CX** como contador no lugar do **registrador BL**; e
2. a instrução **LOOP INICIO** no lugar das instruções **DEC CX** e **JNZ INICIO**.

Ex:

```
TITLE For
.MODEL SMALL
.STACK 100h
.CODE
; Inicializa o contador (neste caso o registrador CX) com o valor 50
MOV CX,50
; Define um rótulo chamado INICIO
INICIO:
; Exibe o caracter "*" na tela
MOV AH,2
MOV DL, "*"
INT 21h
; Decrementa o contador (CX) e:
; caso o contador (CX) seja diferente de zero: salta para INICIO
; caso o contador (CX) seja igual a zero: continua a execução sequencialmente
LOOP INICIO
; Finaliza o programa
MOV AH,4Ch
INT 21h
END
```

Parte 3 - Programa: LETRAS.ASM

Programa que exibe todas as letras maiúsculas na tela:

- 1) Crie a pasta **C:\Temp\OC**
- 2) Faça o download dos arquivos **TASM.EXE** e **TLINK.EXE** na pasta **C:\Temp\OC**
- 3) Abra o **Bloco de Notas** (ou o **JEdit** / ou o **Notepad++**) e digite o programa a seguir:

```
TITLE Letras
.MODEL SMALL
.STACK 100h
.CODE
    ; Inicializa o contador (CX) com o valor 26 (quantidade de letras)
    MOV CX,26
    ; Inicializa o registrador BL com o código ASCII da letra "A" (código ASCII: 65)
    MOV BL,65

    ; Define um rótulo chamado INICIO
    INICIO:
    ; Exibe na tela a letra armazenada em BL
    MOV AH,2
    MOV DL, BL
    INT 21h
    ; Incrementa o valor de BL (muda para a próxima letra)
    INC BL

    ; Decrementa o contador (CX) e:
    ; caso o contador (CX) seja diferente de zero: salta para INICIO
    ; caso o contador (CX) seja igual a zero: continua a execução sequencialmente
    LOOP INICIO

    ; Finaliza o programa
    MOV AH,4Ch
    INT 21h
END
```

- 4) Salve o arquivo com o nome **LETRAS.ASM** na pasta **C:\Temp\OC**
- 5) Abra o **DOS Box**
- 6) Execute o comando: **mount C C:\Temp\OC**
- 7) Execute o comando: **C:**
- 8) Execute comando: **DIR** (verifique se os arquivos **TASM.EXE**, **TLINK.EXE** e **LETRAS.ASM** estão na pasta atual)
- 9) Para compilar o programa, execute o comando: **TASM LETRAS.ASM**
- 10) Execute o comando: **DIR** (verifique se o arquivo **LETRAS.OBJ** foi criado com sucesso)
- 11) Para gerar um executável, execute o comando: **TLINK LETRAS.OBJ**
- 12) Execute o comando: **DIR** (verifique se o arquivo **LETRAS.EXE** foi criado com sucesso)
- 13) Execute o programa com o comando: **LETRAS.EXE**

Atividade para entrega

Crie um programa em linguagem assembly chamado **ATIV4.ASM** que exibe todas as letras minúsculas na tela, exibindo 4 letras por linha.

Exemplo:

```
C:\> ATIV4.EXE
```

```
abcd
```

```
efgh
```

```
ijkl
```

```
mnop
```

```
qrst
```

```
vwxx
```

```
yz
```

ENTREGA

Cada aluno deve:

- 1) Criar uma pasta em seu escaninho no AVA com o nome **Atividade4**.
- 2) Postar o arquivo **ATIV4.ASM** dentro da pasta **Atividade4**.