



## 1 Objetivo

Neste laboratório será vista a estrutura geral de um programa em linguagem de montagem do 8086. Também serão vistos e praticados alguns serviços do DOS para entrada e saída.

## 2 Preliminar

### 2.1 Programa fonte esqueleto

Um programa executável (EXE) rodando sobre o sistema operacional DOS contém 3 segmentos básicos: **code** (código), **data** (dado) e **stack** (pilha). Um esqueleto típico para um programa EXE no NASM é apresentado na figura 1.

---

```
1 SEGMENT code                ; segmento de código
2 ..start:                    ; ponto de entrada do programa
3
4   mov  ax, data              ; inicializa DS
5   mov  ds, ax
6
7   mov  ax, stack             ; inicializa pilha (SS:SP)
8   mov  ss, ax
9   mov  sp, stacktop
10
11 ;
12 ; SEU CODIGO A PARTIR DAQUI
13 ;
14
15   mov  ah, 0x4C              ; volta controle ao DOS
16   int  0x21
17
18
19 SEGMENT data                ; segmento de dados
20 ; SEUS DADOS A PARTIR DAQUI
21
22
23 SEGMENT stack stack        ; segmento de pilha
24   RESB 256
25 stacktop:
```

---

Figura 1: Esqueleto para programas 8086 (NASM)

Os segmentos são identificados pela diretiva `SEGMENT` do montador NASM. As linhas 1 a 16 da figura 1 formam o segmento de código do programa esqueleto. O segmento de dados declarado na linha 19 está vazio, enquanto que o de pilha se estende da linha 23 a 25. É necessário informar ao montador a partir de qual ponto o programa vai começar sua execução. O NASM usa a palavra `..start:` para tal feito. Código de inicialização pode ser necessário caso o programa vá fazer uso de

variáveis do segmento de dados (linhas 4 e 5) ou pilha (linhas 7,8 e 9). O código de finalização que devolve o controle ao DOS deve SEMPRE ser o último código a ser executado dentro do segmento (linhas 15 e 16).

Você pode utilizar o programa esqueleto apresentado na figura 1 como base para seus programas daqui em diante (o arquivo pode ser baixado na página do curso). Comece a adicionar suas instruções logo após a linha 13 (comentário `SEU CODIGO A PARTIR DAQUI`) e declare suas variáveis a partir da linha 20 (comentário `SEUS DADOS A PARTIR DAQUI`).

## 2.2 Serviços do DOS

O sistema operacional DOS possibilita o uso de uma grande quantidade de rotinas já prontas a partir da `INT 0x21`. A grande maioria dos serviços realizam alguma operação de entrada/saída, como leitura a partir do teclado e escrita no vídeo. A escolha do serviço é feita pelo valor contido no registrador de 8 bits `AH`. Dependendo do serviço, alguns registradores podem armazenar parâmetros extras.

Para executar um serviço do DOS você deve setar o registrador `AH` com o número do serviço, setar outros registradores (caso necessário) e executar a instrução `INT 0x21`. A seguir estão listados alguns serviços do DOS para leitura de teclado e escrita no vídeo. Você vai precisar usá-los para as atividades neste laboratório (e muito provavelmente em laboratórios futuros).

### 01 -Entrada pelo teclado (com eco no vídeo)

Entrada: `AH` = 1

Retorno: `AL` = caracter lido (ASCII)

### 02 -Imprime caracter no vídeo

Entrada: `AH` = 2, `DL` = caracter (ASCII)

### 09 - Imprime String

Entrada: `AH` = 9, `DS:DX` = endereço da string (terminada com '\$')

## 3 Atividade 1

Faça um programa em linguagem de montagem que leia um valor hexa do teclado e o imprima em binário. O programa deve fazer a validação da entrada, ou seja, se for digitado algo diferente da faixa de valores 0-f, uma mensagem de erro (por exemplo, `número inválido`) deve ser mostrada.

Você pode usar o programa esqueleto como base (arquivo `skel.asm` na página deste laboratório) e os serviços do DOS apresentados na seção anterior.

## 4 Atividade 2

Faça um programa em linguagem de montagem que leia um valor decimal na faixa 00-99 pelo teclado e imprima sua representação em hexadecimal e binário. Um exemplo para a saída do programa caso seja digitado 33 é:

```
0x21 - 0010 0001
```

Novamente, faça a validação da entrada, apresentando uma mensagem de erro caso seja inválida.