

Aula 5

Instruções de controle

Instruções de controle

- **jmp ENDERECO** (desvio incondicional)

```
.section .text
.globl _start
_start:
    ...
    ...
    ...
    jmp LABEL
    ...
    ...
    ...
LABEL:
    movl $0, %eax
    ...
    ...
```

Instruções de controle

- *Jcc (notem, o desvio é baseado na condição armazenado nos registrador **flags**)*
- *cc*
 - [n]a[e] - acima (não sinalizado maior que). [n] pode ser adicionado para "not" e [e] pode ser adicionado para "ou igual"
 - [n]b[e] - abaixo (não sinalizado menor que)
 - [n]e - igual
 - [n]z - zero
 - [n]g[e] - maior que (sinalizado)
 - [n]l[e] - menor que (sinalizado)
 - [n]c - carry flag set
 - [n]o - overflow flag set
 - [p]p - parity flag set
 - [n]s - sign flag set
 - ecxz - %ecx is zero

Instruções de controle

Exemplo:

```
    cmpl %ax, %bx
    je IGUAL
    ...
    ...
IGUAL:
    ...
    ...
```

- desvio se ax e bx forem iguais
- jg: desvio se bx for maior que ax
- jl: desvio se bx for menor que ax

Instruções de controle

```
if (a < b)                movl a, %eax
    a= 0;                 cmp %eax, b
else                       jg B_MAIOR
    a= 50;                mov $50, a
endif                      jmp CONVERGE
b= 20;

B_MAIOR:
    mov $0, a
CONVERGE:
    mov $20, b
```

Instruções de controle – loop

```
mov 10, %cx
loop_label:
    ...
    ...
    ...
loop loop_label (automaticamente decrementa cx
e testa se igual a zero)
```

*utilize a opção: `as teste.s -gstabs -o teste.o` para facilitar a visualização do código