

Instruções de Desvios Condicionais

Disciplina: Arquitetura de Computadores

Diretiva .globl

► .globl

- Diretiva global
- Semelhante a função principal main() da linguagem C
- Função principal do programa

```
.data                                #Diretiva de dados
    msg: .asciiz                    "Olá mundo!"
.text                                #Diretiva de texto
.globl bloco1                        #Diretiva global
bloco1:                             #Bloco de código Bloco1
    li $v0, 4                       #Impressão de String
    la $a0, msg                     #$a0 recebe msg
    syscall                         #Executa
    li $v0, 10                      #Fim do programa
    syscall                         #Executa
```



Instruções de Desvios Condicionais

- ▶ Os códigos terão o bloco de execução principal com uma condicional

Comando	Exemplo	Significado	Tradução
beq	beq \$t0, \$t1, bloco	Se \$t0 for igual a \$t1, execute a partir do rótulo bloco	branch if equal
bne	bne \$t0, \$t1, bloco	Se \$t0 for diferente de \$t1, execute a partir do rótulo bloco	branch if not equal
blt	blt \$t0, \$t1, bloco	Se \$t0 for menor que \$t1, execute a partir do rótulo bloco	branch if less than
bgt	bgt \$t0, \$t1, bloco	Se \$t0 for maior que \$t1, execute a partir do rótulo bloco	branch if greater than
ble	ble \$t0, \$t1, bloco	Se \$t0 for menor ou igual a \$t1, execute a partir do rótulo bloco	branch if less or equal
bge	bge \$t0, \$t1, bloco	Se \$t0 for maior ou igual a \$t1, execute a partir do rótulo bloco	branch if greater or equal

Instruções de Desvios Condicionais

- ▶ beq (*Branch for Equal*)

- ▶ Desvie se for igual

- ▶ Desvia quando o primeiro registrador é igual ao segundo registrador
 - ▶ Pode ser usada para um programa de menu ou para identificar uma condição específica dentro do código

- ▶ Exemplo:

```
beq $t0, $t1, bloco      #desvia se $t0 = $t1
```



Instruções de Desvios Condicionais

- ▶ **bne** (*Branch for Not Equal*)

- ▶ Desvie se não for igual ou desvie se for diferente
 - ▶ Desvia quando o primeiro registrador é diferente do segundo registrador
 - ▶ Podemos usar a **bne** como um caso contrário ao **beq**

- ▶ Exemplo:

```
bne $t0, $t1, bloco      #desvia se $t0 <> $t1
```



Instruções de Desvios Condicionais

- ▶ **bge** (*Branch if Greater or Equal*)

- ▶ Desvia o fluxo de dados se o número contido no registrador 1 for maior ou igual ao do registrador 2

- ▶ Exemplo:

```
bge $t0, $t1, bloco      #desvia se $t0 >= $t1
```

- ▶ O fluxo é desviado para a sub-rotina chamada bloco quando **\$t0** for maior ou igual a **\$t1**



Instruções de Desvios Condicionais

- ▶ blt (*Branch if Less Than*)

- ▶ Desvia o fluxo de dados se o valor contido no registrador 1 for menor do que o valor contido no registrador 2

- ▶ Exemplo:

```
blt $t0, $t1, bloco      #desvia se $t0 < $t1
```

- ▶ O fluxo é desviado para a sub-rotina chamada bloco quando **\$t0** for menor do que **\$t1**



Instruções de Desvios Condicionais

- ▶ `ble` (*Branch if Less or Equal*)
 - ▶ Desvia o fluxo de dados se o número contido no registrador 1 for menor ou igual ao do registrador 2

- ▶ Exemplo:

```
ble $t0, $t1, bloco      #desvia se $t0 <= $t1
```

- ▶ O fluxo é desviado para a sub-rotina chamada bloco quando **\$t0** for menor ou igual a **\$t1**



Instruções de Desvios Condicionais

- ▶ **bgt** (*Branch if Greater or Than*)

- ▶ Desvia o fluxo de dados se o valor contido no registrador 1 for maior do que o valor contido no registrador 2

- ▶ Exemplo:

```
bgt $t0, $t1, bloco      #desvia se $t0 > $t1
```

- ▶ O fluxo é desviado para a sub-rotina chamada bloco quando **\$t0** for maior do que **\$t1**



Instruções de Desvios Condicionais

- ▶ Exemplo usando as instruções beq e bne

CONDIÇÃO	MENSAGEM
Igual a zero (beq)	Foi detectado o número ZERO!
Diferente de zero (bne)	Este número não é zero!



```

1  .data                                #Diretiva de dados
2  pergunta:        .asciiz "Digite um número: "
3  eh_zero:         .asciiz "Foi detectado o Zero!\n"
4  nao_eh_zero:     .asciiz "Este número não é Zero!\n"
5  .text                                #Diretiva de texto
6  li $v0, 4                #Imprime uma string
7  la $a0, pergunta        #Carrega a string
8  syscall                #Executa
9  li $v0, 5                #Lê um número inteiro
10 syscall                #Executa
11 #Grava o número digitado em $t0
12 move $t0, $v0           #$t0 = $v0
13 #Desvios condicionais usando beq e bne
14 beq $t0,0,seforzero     #Se $t0 for 0, executa "seforzero"
15 bne $t0,0,senaoforzero  #Se $t0 não for 0, executa "senaoforzero"
16 #Condição se for zero
17 seforzero:
18 #Imprimir
19 li $v0, 4                #Imprimir uma string
20 la $a0, eh_zero          #Carrega a string
21 syscall                #Executa
22 #Depois finaliza o programa
23 li $v0, 10               #Sair do programa
24 syscall                #Executa
25 #Condição se não for zero
26 senaoforzero:
27 #Imprimir
28 li $v0, 4                #Imprimir uma string
29 la $a0, nao_eh_zero      #Carrega a string
30 syscall                #Executa
31 #Depois finaliza o programa
32 li $v0, 10               #Sair do programa
33 syscall                #Executa

```

O **beq** e o **bne** estão se comportando neste código como um **SE** e um **SENÃO**. Em ambas as condições, depois de mostrarem a mensagem proposta, o programa é finalizado.