

# 2023/1 CCT ARQUITETURA E ORGANIZAÇÃO DE COMPUTADORES

[Painel](#) / [Meus cursos](#) / [Departamento de Ciência da Computação](#) / [Bacharelado em Ciência da Computação](#) / [2023/1 CC](#)  
/ [2023/1 CCT CCI192-04U AOC0004](#) / [Semana 05](#) / [Q02 - Programação MIPS - somar dígitos recursivo](#)

 [Descrição](#)

 [Visualizar envios](#)

## Nota



Revisado em Sunday, 9 Apr 2023, 16:11 por

**Nota:** 10,00 / 10,00

**Redução final:** 0 [1 / 3 -1] ?

**Relatório de avaliação**  [-]

[\[+\]](#) [Summary of tests](#)

 Enviado em Sunday, 9 Apr 2023, 16:10 ( [Baixar](#))

somarDigitosRecursivo.s

```
1 # autor: Kauan Henrique Werlich
2 # disciplina: Arquitetura e Organização de Computadores (AOC)
3 # esse programa é parecido com o do exercício anterior, só com alguns
4 # passos a menos
5 .data
6 texto: # escrever \n após imprimir número
7 .asciiz "\n"
8
9 .text
10 .globl main
11 main:
12 li $v0, 5 # ori $v0,$zero, 5
13 syscall
14 ori $a0, $v0, 0 # $a0 = $v0 => ler número para somar dígitos
15 beq $a0, $zero, end # se número lido for 0, terminar programa
16
17 jal dividePorDez # salta para a função
18
19 ori $a0, $v0, 0 # $a0 = $v0
20 li $v0, 1 # código para imprimir inteiro
21 syscall
22
23 la $a0, texto
24 li $v0, 4 # print \n após o inteiro
25 syscall
26
27 j main # continue lendo outros números
28 end:
29 li $v0, 10 # código para terminar o programa
30 syscall
31 dividePorDez:
32 beq $a0, $zero, fatorParada # se $a0 for igual a 0, vá para o branch
33
34 addi $sp, $sp, -8 # deslocando o topo da pilha 8 bytes
35 sw $s0, 0($sp) # armazenando $ra nos últimos 4 bytes da pilha
36 sw $ra, 4($sp) # armazenando $s0 nos 4 bytes após $ra
37
38 divu $a0, $a0, 10 # divide $a0 por 10 e põe o quociente em $a0.
39 mfhi $s0 # move o conteúdo de HI (resto) para $s0
40
41 jal dividePorDez # recursão
42
43 add $v0, $s0, $v0 # somar os valores armazenados na pilha
44
45 # restaurar
46 lw $s0, 0($sp) # restaura valor de $s0
47 lw $ra, 4($sp) # restaura valor de $ra
48 addi $sp, $sp, 8 # ajustando topo da pilha para "excluir" os itens
49
50 jr $ra # vai para o endereço salvo em $ra
51
52 fatorParada: # se $a0 for igual a 0, vá para o último endereço salvo
53 add $v0, $zero, $zero
54 jr $ra
```



[VPL](#)

Você acessou como KAUAN HENRIQUE WERLICH (Sair)

2023/1\_CCT\_CCI192-04U\_AOC0004

Português - Brasil (pt\_br)

Deutsch (de)

English (en)

Español - Internacional (es)

Français (fr)

Italiano (it)

Português - Brasil (pt\_br)

日本語 (ja)

Resumo de retenção de dados

Obter o aplicativo para dispositivos móveis

