

12	13	14	15	16
1	.data			
2	vetor: .word	1, 2, 3, 4, 5, 6		
3				
4	.text			
5	.globl	main		
6				
7	main:			
8	la \$a0, vetor	# Endereço do vetor		
9	# Configurar o índice do início e do final			
10	li \$t1, 0	# índice de início do vetor		
11	li \$t2, 20	# índice de final do vetor		
12				
13	loop:			
14	# Verifique se os índices se cruzaram			
15	bge \$t1, \$t2, fim			
16	# Atualiza os endereços dos índices			
17	add \$t5, \$a0, \$t1	# endereço do início		
18	add \$t6, \$a0, \$t2	# endereço do final		
19	# Carrega os elementos do vetor			
20	lw \$t3, 0(\$t5)			
21	lw \$t4, 0(\$t6)			
22	# Troca os elementos			
23	sw \$t4, 0(\$t5)			
24	sw \$t3, 0(\$t6)			
25	# Atualiza os índices			
26	addi \$t1, \$t1, 4			
27	subi \$t2, \$t2, 4			
28	j loop			
29				
30	fim:			
31	li \$v0, 10			
32	syscall			

12	13	14	15	16
1	.data			
2	numero1: .word	36	# Primeiro número	
3	numero2: .word	48	# Segundo número	
4	mdc: .word	0	# Armazena o MDC aqui	
5				
6	.text			
7	.globl	main		
8				
9	main:			
10	lw \$s0, numero1	# Carrega o primeiro número em \$s0		
11	lw \$s1, numero2	# Carrega o segundo número em \$s1		

```

12
13     jal calcula_mdc
14
15     li $v0, 10          # Código de saída
16     syscall
17
18 calcula_mdc:
19     loop:
20         beq $s0, $s1, fim # Se os números forem iguais, termina
21         bgt $s0, $s1, caso1 # Se $s0 > $s1
22         bgt $s1, $s0, caso2 # Se $s1 > $s0
23     caso1:
24         sub $s0, $s0, $s1 # Subtrai $s1 de $s0
25         j loop
26     caso2:
27         sub $s1, $s1, $s0 # Subtrai $s0 de $s1
28         j loop
29     fim:
30         move $s2, $s0     # Move o resultado para $s2
31         sw $s2, mdc       # Armazena o MDC na variável
32         jr $ra            # Retorna
33

```

12	13	14	15	16
1	.data			
2	vetor: .word 11, 52, 33, 64, 95, 26, 7, 48, 29, 10			
3				
4	.text			
5	.globl main			
6				
7	main:			
8	la \$a0, vetor		# Endereço do vetor	
9	li \$t0, 0		# Índice	
10	lw \$s0, (\$a0)		# Maior	
11	lw \$s1, (\$a0)		# Menor	
12				
13	loop:			
14	beq \$t0, 10, fim		# Se o índice atingir 10 (tamanho do vetor), vá para o fim	
15	lw \$t1, (\$a0)		# Carrega o valor atual do vetor em \$t1	
16	bge \$t1, \$s0, maior			
17	ble \$t1, \$s1, menor			
18				
19	incremento:			
20	addi \$t0, \$t0, 1		# Incrementa o índice	
21	addi \$a0, \$a0, 4		# Atualiza o endereço para o próximo elemento	
22	j loop			
23				
24	fim:			
25	li \$v0, 10			
26	syscall			
27				
28	maior:			
29	move \$s0, \$t1		# Atualiza o maior	
30	j incremento			
31				
32	menor:			
33	move \$s1, \$t1		# Atualiza o menor	
34	j incremento			
35				

12	13	14	15	16
1	.text			
2	main:			
3	li \$s0, 5		# a	

```

4  add $t0, $s0, $s0    # b = a + a
5  add $t1, $t0, $t0    # c = b + b
6
7  li $a0, 3            # g
8  li $a1, 15           # h
9  li $a2, 4            # i
10 li $a3, 7            # j
11 jal exemplo          # d
12 move $s1, $v0         # move o valor retornado
13 sub $s0, $t1, $t0     # a = c - b
14 add $s0, $s0, $s1     # a = a + d
15 j exit
16
17 exemplo:             # calcula d
18 sub $sp, $sp, 12      #prepara pilha
19 sw $t1, 8($sp)
20 sw $t0, 4($sp)
21 sw $s0, 0($sp)
22 add $t0, $a0, $a1     # (g + h)
23 add $t1, $a2, $a3     # (i + j)
24 sub $s0, $t0, $t1     # d = (g + h) - (i + j)
25 move $v0, $s0         # retorna o valor
26 lw $s0, 0($sp)        # desempilha
27 lw $t0, 4($sp)
28 lw $s0, 8($sp)
29 add $sp, $sp, 12
30 jr $ra               # retorna para logo depois de exemplo
31 exit:
32 li $v0, 10
33 syscall

```

12	13	14	15	16	
1	.text				
2	.globl main				
3					
4	main:				
5	li \$a0, 5	# Coloque o número cujo fatorial você deseja calcular em \$a0			
6	li \$v0, 1	# Inicialize \$v0 com 1 (resultado inicial)			
7	jal fact	# Chame a função fact			
8	move \$a0, \$v0	# \$a0 recebe o resultado da função			
9	li \$v0, 10	# Encerre o programa			
10	syscall	# Retorne ao endereço de retorno			
11					
12	fact:				
13	beqz \$a0, base	# Se \$a0 for igual a zero, vá para o caso base			
14	mul \$v0, \$v0, \$a0	# \$v0 recebe \$v0 * \$a0			
15	subi \$a0, \$a0, 1	# Decrementar \$a0			
16	j fact	# Chame fact recursivamente			
17					
18	base:				
19	jr \$ra				