

```

1      .data
2      line1: .asciiz "Geben Sie die 1. Zahl ein: "
3      line2: .asciiz "Geben Sie die 2. Zahl ein: "
4      erg:   .asciiz "Das Ergebnis lautet: "
5      komma: .asciiz ", "
6
7      # Registerbelegungen
8      # $t0 = A
9      # $t1 = B
10     # $t2 = Summe (von A und B)
11     # $t3 = 2
12     # $t4 = Rest der Division
13     # $t5 = 10
14     # $s0 = Vorkommazahl
15     # $s1 = Nachkommazahl
16
17     .text
18
19 main:
20     #####
21     # Einlesen der 2 Werte
22     #####
23
24     li      $v0, 4          # 4: print_str:
25     la      $a0, line1      # Adresse der 1. auszugebende Zeile nach $a0
26     syscall
27
28     li      $v0, 5          # 5: read_int
29     syscall
30     move     $t0, $v0       # eingelesenen Wert in Register $t0 verschieben (A)
31
32     li      $v0, 4          # 4: print_str:
33     la      $a0, line2      # Adresse der 2. auszugebende Zeile nach $a0
34     syscall
35
36     li      $v0, 5          # 5: read_int
37     syscall
38     move     $t1, $v0       # eingelesenen Wert in Register $t1 verschieben (B)
39
40     #####
41     # Berechnung des Durchschnitts
42     #####
43
44     add      $t2, $t0, $t1   # t2/Summe = $t0/A + $t1/B
45     li      $t3, 2          # Den Wert 2 in Register $t0 laden
46     div      $s0, $t2, $t3   # $s0/Vorkomma = $t2/Summe DIV $t3/2 (whole number divide)
47     rem      $t4, $t2, $t3   # $t4/Rest = $t2/Summe MOD $t3/2
48
49     { li      $t5, 10        # Den Wert 10 in Register $t5 laden
50       mul     $t6, $t4, $t5  # Eine Null an den Rest haegen (= x/10) }
51
52     div      $s1, $t6, $t3   # den mit 0 erweiterten Rest wieder durch 2 teilen = NKS
53
54     #####
55     # Ausgabe des Ergebnisses
56     #####
57
58     li      $v0, 4          # 4: print_str

```

1)

2)

3)

$79 \div 2 = 39,5$
 $2 \overline{) 79}$
 $\underline{6}$
 19
 $\underline{18}$
 10
 $\underline{10}$
 0

```

59      la      $a0, erg      # Adresse des Ergebnis-Texts nach $a0
60      syscall              # ausgeben
61
62      li      $v0, 1        # 1: print_int
63      move    $a0, $s0      # Vorkommastelle nach la0
64      syscall
65
66      li      $v0, 4        # 4: print_str
67      la      $a0, komma    # Adresse des Komma-Texts nach $a0
68      syscall
69
70      li      $v0, 1        # 1: print_int
71      move    $a0, $s1      # Nachkommastelle nach $a0
72      syscall
73
74      li      $v0, 10       # Systemaufrufnr. 10 = EXIT
75      syscall

```