

**Aufgabe 1: Schleifen**

Sie haben folgendes Code-Fragment in einer Hochsprache gegeben:

```
int sum = 0;
for(i = 1; i < 100; i = i * 2)
    sum = sum + i;
```

Geben Sie eine entsprechende Sequenz von MIPS-Befehlen an, die das Code-Fragment realisiert. Beachten Sie dabei:

- Der Wert der Variable  $i$  wird im Register  $\$s0$  gespeichert.
- Der Wert der Variable  $sum$  wird im Register  $\$s1$  gespeichert.

**Aufgabe 2: Programm verstehen**

Sie haben folgendes Programm gegeben:

```
.data

size:    .word    10
arr:     .word    10, -5, 8, 0, 6, 87, -68, 97, 25, -32

.text

main:    la $s0, size
         lw $s1, 0($s0)
         ori $s2, $zero, 0
         ori $s3, $zero, 0
         ori $s4, $zero, 0
         ori $s5, $zero, 0
         la $s6, arr

loop:    bge $s5, $s1, done
         lw $s7, 0($s6)
         addu $s2, $s2, $s7
         blez $s7, lez
         addu $s3, $s3, $s7
lez:     bgez $s7, inc
         addu $s4, $s4, $s7
inc:     addi $s5, $s5, 1
         addi $s6, $s6, 4
         j loop

done:    li $v0, 10
         syscall
```

Beschreiben Sie, welche Funktionalität durch das Programm realisiert wird. Gehen Sie dabei auf folgende Punkte ein:

- Welche Datenstruktur wird hier verwendet?
- Wie viele Schleifen finden Sie in diesem Programm?
- Welche drei Werte werden in diesem Programm berechnet? Wie werden diese Werte berechnet?

### **Aufgabe 3: Unterprogramm**

Sie haben folgendes Unterprogramm gegeben:

```
func:  sll $t1, $a1, 2
      add $t1, $a0, $t1
      lw  $t0, 0($t1)
      lw  $t2, 4($t1)
      sw  $t2, 0($t1)
      sw  $t0, 4($t1)
      jr  $ra
```

Beantworten Sie bitte folgende Fragen:

- Wie viele Parameter werden übergeben?
- Was macht dieses Unterprogramm? Hinweis: in \$a0 steht die Basisadresse eines Arrays.
- Warum müssen hier keine Werte auf den Stack gerettet werden?

### **Aufgabe 4: Unterprogramm**

Sie haben folgende Implementierung eines Unterprogramms in MIPS gegeben:

```
func:  move $v0, $zero
loop:  beq $v0, $a1, done
      sll $t0, $v0, 2
      add $t0, $t0, $a0
      lw  $t0, 0($t0)
      bne $t0, $a2, skip
      jr  $ra
skip:  addi $v0, $v0, 1
      b  loop
done:  li  $v0, -1
      jr  $ra
```

Welche Funktionalität wird durch das Unterprogramm func implementiert? Geben Sie einen entsprechenden Pseudocode (oder auch C oder Java-Code) an, der dem obigen Code-Fragment entspricht. Hinweis: in \$a0 steht die Basisadresse eines Arrays.

### **Aufgabe 5: MIPS-Programm**

Implementieren Sie im MARS-Simulator folgendes MIPS-Programm:

- Ihr Programm soll am Anfang 5 Integer-Werte einlesen und im Speicher ablegen.
- Die Werte im Speicher sollen danach in umgekehrter Reihenfolge (letzter eingelesener Integer-Wert zuerst) ausgegeben werden.

Sie dürfen Funktionalität in Unterprogramme auslagern!