

Vorname

Nachname

Matrikel-Nr

Diese Klausur ist mein **dritter Prüfungsversuch** (bitte ankreuzen): Ja ☐ Nein ☐

Schreiben Sie jede Lösung möglichst auf die Vorderseite eines *neuen Blattes* (und lassen Sie die Rückseiten Ihrer Lösungsblätter *leer*).

Aufgabe 1 (15 Punkte): Schreiben Sie ein Unterprogramm entsprechend der folgenden Spezifikation:

```
1      static public int anzDoppels(String[] sr) {
2          // Wie oft kommen in den String-Komponenten der Reihung sr zwei
3          // kleine s ("ss") nacheinander vor? Diese Funktion liefert die
4          // Antwort. Beispiele:
5          // String[] srA = {"s", "ss", "sss", "ssss"};
6          // String[] srB = {"assel", "bessel", "cassel"};
7          // String[] srC = {"abc", "", "def"};
8          // String[] srD = {};
9          //
10         // anzDoppels(srA) ist gleich 4
11         // anzDoppels(srB) ist gleich 3
12         // anzDoppels(srC) ist gleich 0
13         // anzDoppels(srD) ist gleich 0
14         ...
15     } // anzDoppels
16
```

Aufgabe 2 (15 Punkte): Schreiben Sie ein Unterprogramm entsprechend der folgenden Spezifikation:

```
1      static public int anzNullBits(int n) {
2          // Wieviele Bits von n haben den Wert 0? Diese Funktion liefert
3          // die Antwort. Beispiele:
4          // anzNullBits(0x00000000) ist gleich 32
5          // anzNullBits(0xFFFFFFFF) ist gleich 0
6          // anzNullBits(0x00000001) ist gleich 31
7          // anzNullBits(0x80000000) ist gleich 31
8          // anzNullBits(0xFFFF0000) ist gleich 16
9          ...
10     } // anzNullBits
```

Aufgabe 3 (15 Punkte): Betrachten Sie die folgenden drei Befehlsfolgen (Teilaufgaben):

```
1      // Teilaufgabe 3.1.:
2      int a = -3;
3      int b = +8;
4      for (int i=a; i<=b; i++) {
5          a++;
6          b--;
7          if (i%2 == 0) i++;
8          pln(i + ", " + b);
9      }
10
11     // Teilaufgabe 3.2.:
12     char c = 'A'; // Unicode-Codezahl fuer das Zeichen A: 65
13     int d = 'A';
14     while (true) {
15         pln(c + ": " + d);
16         if (d > 70) break;
17         c += 3;
18         d += 3;
19     }
```

```

20
21      // Teilaufgabe 3.3.:
22      for (int i=1; i<=5; i++) {
23          for (int j=0; j<i; j++) {
24              p("a");
25          }
26          pln("b");
27      }

```

Geben Sie für jede der drei Befehlsfolgen an, welche Zeile(n) sie zum Bildschirm ausgibt.

Aufgabe 4 Betrachten Sie die folgenden Klassenvereinbarungen:

```

1  class Schulz {
2      static int a;
3      static int b;
4
5      int c;
6
7      Schulz(int c) {
8          this.c = c;
9          a += 1;
10         b += c;
11     }
12
13     static public void main(String[] _) {
14         ...
15         Schulz anna = new Schulz(3);
16         ...
17         Schulz bert = new Schulz(5);
18         ...
19         Schulz carl = new Schulz(1);
20         ...
21     } // main
22
23 } // class Schulz

```

Stellen Sie sich die vier *Zeitpunkte* vor, an denen der Ausführer die Zeile 14, 16, 18 bzw. 20 erreicht. Beantworten Sie die folgenden Fragen für jeden dieser vier Zeitpunkte einmal:

Fragen (Was gilt in diesem Moment?)	Zeile 14	Zeile 16	Zeile 18	Zeile 20
Wieviele <i>Module</i> gibt es?				
Wieviele int- <i>Variablen</i> gibt es?				
Wieviele int-Variablen namens a gibt es?				
Wieviele int-Variablen namens c gibt es?				
Welchen <i>Wert</i> hat die Variable Schulz.a?				
Welchen <i>Wert</i> hat die Variable Schulz.b?				

Aufgabe 5 (15 Punkte): Betrachten Sie die folgenden Vereinbarungen von drei Variablen:

```
1      ...
2      String    s1 = "Hallo!";
3      String    s2 = "Sonja!";
4      String[]  sr = {s1, s2, s1};
5      ...
```

Stellen Sie diese Variablen als Bojen dar. Es ist erlaubt, die *vereinfachte* Bojendarstellung zu benutzen. Wenn Sie sich nicht sicher sind, was damit gemeint ist, sollten Sie wahrscheinlich die ausführliche Bojendarstellung benutzen (die ist immer erlaubt).

Aufgabe 6: (15 Punkte) Vereinbaren Sie ein Klasse namens `Karl` und darin die folgenden Elemente und Konstruktoren:

1. Ein *unveränderbares Klassenattribut* namens `a1` vom Typ `String`.
2. Ein *veränderbares Objektattribut* namens `a2` vom Typ `int`.
3. Einen *Standardkonstruktor*, der `a2` initialisiert.
4. Einen *weiteren Konstruktor*, der ebenfalls `a2` initialisiert.
5. Eine *Objektmethode* namens `m1` mit zwei `long`-Parametern und dem Ergebnistyp `boolean`.
6. Eine *Klassenprozedur* namens `m2` mit einem `String`-Parameter.

Alle nicht ausdrücklich festgelegten Eigenschaften der Elemente und Konstruktoren dürfen Sie *beliebig gestalten*. Ihre Vereinbarung der Klasse `Karl` soll aber *keine formalen Fehler* enthalten (d. h. sie soll *ohne Fehlermeldungen compilierbar* sein).

Lösung 1 (15 Punkte):

```
1 static public int anzDoppels(String[] sr) {
2     // Wie oft kommen in den String-Komponenten der Reihung sr zwei
3     // kleine s ("ss") nacheinander vor? Diese Funktion liefert die
4     // Antwort.
5
6     int anz = 0;
7     for (String s: sr) {
8         for (int i=0; i<s.length()-1; i++) {
9             if (s.substring(i, i+2).equals("ss")) {
10                anz++;
11                i++;
12            }
13        } // for i
14    } // for s
15    return anz;
16 } // anzDoppels
```

Lösung 2 (15 Punkte):

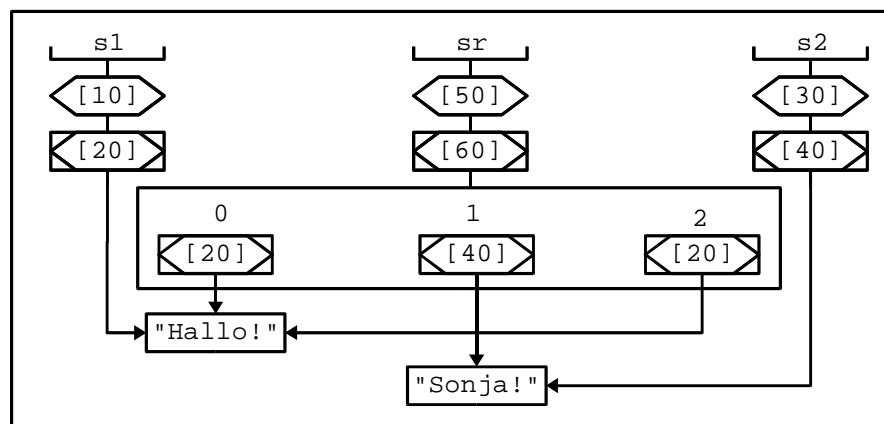
```
1 static public int anzNullBits(int n) {
2     // Wieviele Bits von n haben den Wert 0? Diese Funktion liefert
3     // die Antwort.
4
5     int anz = 0;
6     for (int i=1; i<=32; i++) {
7         if (n%2 == 0) anz++;
8         n = n >> 1;
9     }
10    return anz;
11 } // anzNullBits
```

Lösung 3 (15 Punkte):

```
1 3.1. ....
2
3 -3, 7
4 -1, 6
5 1, 5
6 3, 4
7 5, 3
8
9 3.2. ....
10
11 A: 65
12 D: 68
13 G: 71
14
15 3.3. ....
16
17 ab
18 aab
19 aaab
20 aaaab
21 aaaaab
```

Lösung 4 (15 Punkte):

Fragen (Was gilt in diesem Moment?)	Zeile 14	Zeile 16	Zeile 18	Zeile 20
Wiviele <i>Module</i> gibt es?	1	2	3	4
Wiviele <i>int-Variablen</i> gibt es?	2	3	4	5
Wiviele <i>int-Variablen</i> namens <i>a</i> gibt es?	1	1	1	1
Wiviele <i>int-Variablen</i> namens <i>c</i> gibt es?	0	1	2	3
Welchen <i>Wert</i> hat die Variable <i>Schulz.a</i> ?	0	1	2	3
Welchen <i>Wert</i> hat die Variable <i>Schulz.b</i> ?	0	3	8	9

Lösung 5 (15 Punkte): Die Variablen *s1*, *s2* und *sr* in vereinfachter Bojendarstellung**Lösung 6 (15 Punkte): Die Klasse Karl**

```

1  class Karl {
2      static final String a1 = "Hallo!";
3
4          int    a2;
5
6      Karl()    {a2 = 17;}
7
8      Karl(int n) {a2 = n;}
9
10         boolean m1(long n1, long n2) {return n1 < n2;}
11     static void  m2(String s)         {System.out.println(s);}
12
13 } // class Karl

```