Universität Mannheim
Lehrstuhl für Praktische Informatik IV
Professor Dr. W. Effelsberg



# Übungsblatt 1 – Musterlösung

## *Aufgabe 1: Rechneraufbau [10 Punkte]*

(a) [7 Punkte]

Das angegebene Programm addiert die Zahlen zusammen, die im Speicherbereich liegen, der durch die Speicherzellen 3 und 4 angegeben wird. Das Ergebnis wird in Speicherzelle 2 gespeichert.
Mit der angegeben Speicherbelegung addiert das Programm den Inhalt der Speicherzellen 5 bis 10.

| Schritt | Akkumulator | Zelle 1 | Zelle 2 | Zelle 3 | Zelle 4 |
|---|---|---|---|---|---|
| vor Beginn | ? | 1 | 0 | 5 | 10 |
| 1. LDI 3 | **12** | **1** | **0** | **5** | **10** |
| 2. ADD 2 | **12** | **1** | **0** | **5** | **10** |
| 3. STA 2 | **12** | **1** | **12** | **5** | **10** |
| 4. LDA 3 | **5** | **1** | **12** | **5** | **10** |
| 5. SUB 4 | **-5** | **1** | **12** | **5** | **10** |
| 6. JMZ 11 | **-5** | **1** | **12** | **5** | **10** |
| 7. LDA 3 | **5** | **1** | **12** | **5** | **10** |
| 8. ADD 1 | **6** | **1** | **12** | **5** | **10** |
| 9. STA 3 | **6** | **1** | **12** | **6** | **10** |
| 10. JMP 1 | **6** | **1** | **12** | **6** | **10** |
| 11. LDI 3 | **7** | **1** | **12** | **6** | **10** |
| 12. ADD 2 | **19** | **1** | **12** | **6** | **10** |
| 13. STA 2 | **19** | **1** | **19** | **6** | **10** |
| 14. LDA 3 | **6** | **1** | **19** | **6** | **10** |
| 15. SUB 4 | **-4** | **1** | **19** | **6** | **10** |
| 16. JMZ 11 | **-4** | **1** | **19** | **6** | **10** |
| 17. LDA 3 | **6** | **1** | **19** | **6** | **10** |
| 18. ADD 1 | **7** | **1** | **19** | **6** | **10** |
| 19. STA 3 | **7** | **1** | **19** | **7** | **10** |
| 20. JMP 1 | **7** | **1** | **19** | **7** | **10** |
| ... | ... | ... | ... | ... | ... |
| 57. HLT | 0 | 1 | 68 | 10 | 10 |

(Für die Lösung der Aufgabe waren nur die Zeilen 1.-20. verlangt.)

(b) [3 Punkte]

Berechnung der Anzahl der Schritte, wenn in der Speicherzelle 4 zu Beginn des Programms $n$ steht:
(Anmerkung: es wird davon ausgegangen, dass $n \geq 5$, da anderenfalls das Programm nicht terminiert!)
Solange der Inhalt der Speicherzelle $3 < n$ ist, werden die Schritte 1-10 ausgeführt:

$$(n - 5) \cdot 10 \quad \text{Schritte}$$

Im letzten Durchlauf (Speicherzelle $3 = n$) werden die Schritte 1-6 und 11 ausgeführt:

$$7 \quad \text{Schritte.}$$

Macht zusammen:

$$\mathbf{(n - 5) \cdot 10 + 7} \quad \text{Schritte.}$$

In der gegebenen Speicherbelegung ist $n = 10$, also werden

$$\mathbf{57} \quad \text{Schritte}$$

ausgeführt.

## *Aufgabe 2: Java–Programmierung [10 Punkte]*

(a) [3 Punkte]

Application – Ausgabe mit `println()`:
Um ein „A“ auszugeben, könnte der Inhalt der Datei `Aufgabe12a.java` ungefähr so aussehen:

```
/* Uebungsblatt 1 - Aufgabe 2 a) */

class Aufgabe12a {
     public static void main(String[] args) {
           System.out.println("  **  ");
           System.out.println(" *  * ");
           System.out.println("*    *");
           System.out.println("******");
           System.out.println("*    *");
           System.out.println("*    *");
     }
}
```

Kompiliert wird das Programm dann durch den Aufruf von `javac Aufgabe12a.java`, anschließend wird es durch `java Aufgabe12a` gestartet.

(b) [3 Punkte]

Application – Ausgabe mit `drawString()` in einem Frame:

```
/* Uebungsblatt 1 - Aufgabe 2 b) */

import java.awt.*;

class Aufgabe12b extends Frame{
     public void paint(Graphics g) {  // Die Methode paint von Java
                                      // automatisch aufgerufen
           g.drawString("******",60,50);
           g.drawString("*     ",60,60);
           g.drawString("*     ",60,70);
           g.drawString("****  ",60,80);
           g.drawString("*     ",60,90);
           g.drawString("*     ",60,100);
           g.drawString("******",60,110);
     }
     Aufgabe12b() {
           super("Aufgabe 2 b");  // Setzt den Fenstertitel
           setSize(160,120);      // Legt die Größe des Fensters fest
           setVisible(true);      // Zeigt das Programmfenster an
     }
     public static void main(String[] args) {
           new Aufgabe12b();      // Ein neues Fenster erzeugen
     }
}
```

Wenn das Programm mit `java Aufgabe12b` aufgerufen wird (zuvor kompilieren!), öffnet sich ein neues Fenster. Das Programm läßt sich nur durch drücken von `Strg+C` in der Konsole wieder beenden, nicht durch die Schaltflächen des Fensters.
(Hinweis: Je nach Buchstabe, sieht das Ergebnis eventuell nicht wie erwartet aus, da Java standardmäßig eine Proportionalschrift verwendet, und deshalb die verschiedenen Zeilen nicht unbedingt korrekt untereinander stehen.)

(c) [4 Punkte]

ca) Applet – Ausgabe mit `println()`:

Um das Applet starten zu können, benötigt man ausser der durch kompilieren erzeugten `Aufgabe12ca.class` auch noch eine HTML–Datei, mit der das Applet im Browser oder AppletViewer geladen werden kann. Diese Datei ist als `Aufgabe12ca.html` ebenfalls abgedruckt.

Das Applet wird im AppletViewer durch Eingabe von `appletviewer Aufgabe12ca.html` auf der Konsole gestartet. Die Ausgabe erfolgt auf der Konsole von der aus der Start erfolgte.

Zum Starten des Applets im Browser, muss die Datei `Aufgabe12ca.html` in den Browser geladen werden. Die Ausgabe erfolgt dann auf der Java-Konsole des Browsers. Die Java-Konsole kann im englischen Netscape Communicator über den Menüaufruf: *Communicator*|*Tools*|*Java Console* erreicht werden.

Datei `Aufgabe12ca.java`:

```
/* Uebungsblatt 1 - Aufgabe 2 ca) */

import java.applet.*;

public class Aufgabe12ca extends Applet{
     public void init() {
           System.out.println("******");
           System.out.println("*     ");
           System.out.println("*     ");
           System.out.println("****  ");
           System.out.println("*     ");
           System.out.println("*     ");
           System.out.println("******");
     }
}
```

Datei `Aufgabe12ca.html`:

```
<html>
     <head>
           <title>Aufgabe 2ca</title>
     </head>
     <body>
           <applet code=Aufgabe12ca width=160 height=120></applet>
     </body>
</html>
```

cb) Applet – Ausgabe mit `drawString()`:

Datei `Aufgabe12cb.java`:

```
/* Uebungsblatt 1 - Aufgabe 2 cb) */

import java.awt.*;
import java.applet.*;

public class Aufgabe12cb extends Applet{
     public void paint(Graphics g) {
           g.drawString("******",60,50);
           g.drawString("*     ",60,60);
           g.drawString("*     ",60,70);
           g.drawString("****  ",60,80);
           g.drawString("*     ",60,90);
           g.drawString("*     ",60,100);
           g.drawString("******",60,110);
     }
}
```

Zum Start des Applets wird wiederum eine HTML–Datei benötigt. Diese ist jedoch fast identisch mir der zu (ca), deshalb wird sie nicht nochmals abgeruckt.