

Algorithmen, Datenstrukturen und Programmieren I

WS 2002/2003

- 1* Fertigen Sie ein Struktogramm an, das den Vorgang "Geld abheben an einem Bankomaten" beschreibt.
2. Welches Ergebnis liefern die folgenden Operationen? Geben Sie zunächst einen "educated guess" an und überprüfen Sie diesen dann mittels einer selbstgeschriebenen Java-Applikation.
 - a. $x = 7 + 3 * 6 / 2 - 1;$
 - b. $x = 2 \% 2 + 2 * 2 - 2 / 2;$
 - c. $x = (3 * 9 * (3 + (9 * 3 / (3))));$
3. Stellen Sie analog zu Beispiel 2. fest, welchen Wert jeweils die Variablen a, b am Ende der angegebenen Anweisungsfolge haben, wenn beide zunächst mit dem Wert 1 initialisiert wurden.
 - a. $b = a++;$
 - b. $b = a--;$
 - c. $b = ++a + b--;$
 - d. $a = b /= 3;$
 - e. $a = (--a + 1) == b ? 4 : 3;$
 - f. $b = a != b++ ? b++ : ++a;$
 - g. $b = a++ == b ? b++ : b * a--;$
 - h. $a /= ++b \% (a + 2);$
- 4* Nehmen Sie an, die boolean-Variablen r, s seien mit true bzw. false initialisiert. Welchen Wert haben die folgenden Ausdrücke? Testen Sie Ihre Meinung wieder mit einem Programm.
 - a. $r \ \&\& \ s \ || \ (r \ || \ s)$
 - b. $r = r \ || \ s \ ? \ r \ : \ s;$
 - c. $r \ || \ (r \ \&\& \ s) \ \&\& \ (r \ || \ s)$
 - d. $s = r \ ? \ r \ : \ (r \ || \ s);$
 - e. $r = (!s \ \&\& \ r) \ ? \ true \ : \ !r;$
 - f. $s = true \ ? \ !s \ : \ !r;$

5* Berechnen Sie den Wert der Zahl π mit Hilfe der unendlichen Reihe

$$\pi = 4 - \frac{4}{3} + \frac{4}{5} - \frac{4}{7} + \frac{4}{9} - + \dots$$

Ihr Programm soll eine Tabelle ausgeben, in der π durch einen, zwei, usw. Terme dieser Reihe approximiert wird. Fertigen Sie zuerst ein Struktogramm an.

6. Bestimmen Sie alle pythagoräischen Tripel, also alle natürlichen Zahlen (a, b, c) mit $a^2 + b^2 = c^2$, wobei a, b und c unter 500 liegen sollen und geben Sie das Ergebnis in einer Tabelle aus. Fertigen Sie zuerst ein Struktogramm an.
7. Schreiben Sie eine Applikation, die 5 Zahlen von der Tastatur einliest und dann in 5 Zeilen jeweils so viele * ausgibt, wie diese Zahlen angeben. Fertigen Sie zuerst ein Struktogramm an.
8. Benutzen Sie das *Sieb des Eratosthenes*, um alle Primzahlen ≤ 999 zu bestimmen und auszudrucken:
 - a) Legen Sie einen Array mit 1000 `boolean` Elementen an und initialisieren Sie alle Elemente mit `true`.
 - b) Beginnend mit Array-Index 2 soll das Programm für jedes Array-Element, das auf `true` steht eine Schleife durchlaufen, in der alle Elemente, deren Index ein Vielfaches dieses Index ist, auf `false` gesetzt werden.

Nachdem dieser Prozess abgeschlossen ist, sind die Primzahlen ≤ 999 genau die Indizes der verbleibenden `true`-Elemente. Diese sollen dann formatiert ausgegeben werden. Fertigen Sie zuerst ein Struktogramm an.

9* Schreiben Sie eine Applikation, die eine Folge von 10 ganzen Zahlen von der Tastatur einliest und sie absteigend sortiert wieder ausgibt. Fertigen Sie zuerst ein Struktogramm an.

10. Schreiben Sie eine Applikation `Zahlenraten`, bei der der Benutzer eine vom Computer intern ermittelte (Integer-) Zufallszahl zwischen 0 und 999 durch Eingaben über die Tastatur herausfinden soll. Zufallszahlen generiert man mit Hilfe der Methode `Math.random()`, die eine Zufallszahl vom Typ `double` im Wertebereich $[0, 1)$ retourniert. Die Deklaration

```
z = (int) (1000*Math.random());
```

liefert also eine zufällige Integer-Zahl zwischen 0 und 999. Das Programm soll nach jedem Tipp des Users zu hoch!, zu niedrig! oder richtig! ausgeben. Ein- und Ausgabe des Programmes sollten über ein JOptionPane-Objekt, wie im Beispiel Eingabe_Swing.java gezeigt, erfolgen.

- 11* Überlegen Sie sich einen rekursiven Algorithmus zur Primfaktorzerlegung einer Zahl. Überzeugen Sie sich anhand der Zahlen 1, 2, 7 und 100 davon, daß Ihr Algorithmus korrekt ist. Schreiben Sie daraufhin ein Testprogramm in Java, das eine Zahl von der Tastatur einliest und ihre Primfaktoren ausgibt. Fertigen Sie zuerst ein Struktogramm an.
12. *Das Acht-Damen-Problem.* Schreiben Sie ein Programm, das acht Damen auf einem Schachbrett so positioniert, daß sie sich gegenseitig nicht bedrohen (im Schachspiel beherrscht eine Dame die Spalte, Zeile und die beiden Diagonalen, an deren Schnittpunkt sie sich befindet). Haben Sie eine Lösung gefunden, soll diese formatiert am Bildschirm ausgegeben werden.

Hinweis:

Dieses Problem lässt sich elegant durch einen rekursiven Algorithmus lösen:

```
dameSetzen(spalte,zeile)
beginne
  wenn spalte größer 8
    lösung ausgeben
  beenden
  wenn zeile größer 8
    in der vorigen spalte dame um eine zeile weiter setzen
    position dieser dame speichern
  gibt es eine bedrohung am brett?
    ja: in dieser spalte dame auf die nächste zeile setzen
    nein: in der nächsten spalte in der ersten zeile
        eine dame setzen
ende
```

Um “Gibt es eine Bedrohung?” zu prüfen, schreibt man eine eigene boolean-Methode mit einer Schleife, die von Spalte 1 bis eine vor der aktuellen Spalte nachsieht, ob in derselben Zeile und in den beiden von der Dame ausgehenden Diagonalen schon eine Dame steht.

Die Position der Damen selbst stellt man durch ein Integer Array der Größe 8 dar, der Index ist die Spalte, der Wert die Zeile.

Begonnen wird die Rekursion in der main Methode, indem man eine Dame auf das Feld (1,1) setzt.

Die Ausgabe der gefundenen Lösung soll in der Form "abcdefgh" erfolgen, wo die Zahl a bis h die Spalte angibt, in der jeweils eine Dame aufgestellt wird.

- 13* a) Implementieren Sie eine Klasse für einen Vektor in zwei Dimensionen. Die Schnittstelle soll Funktionen zum Zuweisen und Abfragen der x und y Komponente, darüber hinaus Addition und Subtraktion, Multiplikation mit Vektor und Skalar sowie Normierung des Vektors beinhalten. Außerdem soll die Klasse über eine `static` Methode verfügen, die einen Nullvektor zurückliefert. Überlegen Sie sich die sinnvolle Verwendung von Konstruktoren.
- b) Schreiben Sie eine Applikation, um die Funktionalität ihrer Klasse zu überprüfen. Implementieren Sie dazu in der `Vektor` Klasse eine Methode zur formatierten Ausgabe des Vektors.
14. Schreiben Sie eine Applikation, die einen von der Kommandozeile entgegengenommenen String darauf prüft, ob er ein Palindrom ist. (Ein Palindrom ist ein String, der von vorne und hinten gelesen das gleiche Wort ergibt, z. B. Anna, Otto.) Verwenden Sie dazu eine Methode `stringReverse`, die einen String entgegennimmt und den umgekehrten String zurückliefert. Fertigen Sie zuerst ein Struktogramm an.
- Hinweis:* Das j te Zeichen eines Strings erhalten Sie durch die `String`-Methode `charAt(j)`. Um Groß-/Kleinschreibung zu ignorieren, verwenden Sie die Methode `toLowerCase()` aus `String`, die alle Zeichen des Strings in Kleinbuchstaben umwandelt.
- 15* Definieren Sie eine von `Angestellter` abgeleitete Klasse `Chef`, die Datenfelder zur Speicherung der Vertragsdauer und der geleiteten Abteilungen enthält. Stellen Sie Methoden zur Änderung und Ausgabe dieser Daten zur Verfügung. Modifizieren Sie die Klasse `Personalverwaltung` so, daß für alle Angestellten (also auch für `Chef`) schon bei der Eingabe Monats- bzw. Stundenlohn und Stundenzahl festgelegt werden können. Nach Eingabe aller Angestellten soll außerdem eine gezielte Abfrage der Daten der einzelnen Angestellten möglich sein: Zuerst sollen die Namen und Nummern aller Angestellten ausgegeben werden. Danach sollen durch Eingabe der Angestellten-Nummer alle Daten des jeweiligen Angestellten abgerufen werden können, solange der User weitere

Daten abrufen will. Verwenden Sie für die Entscheidung, welche Methoden zur Ausgabe der Daten verwendet werden sollen, den `instanceof`-Operator.

16. Schreiben Sie ein `JApplet`, das mittels dreier `JTextFields` drei Zahlen einliest. Nach Eingabe der dritten Zahl (und Drücken von Enter) sollen in nicht-editierbaren `JTextFields` Summe, Produkt, Mittelwert, Minimum und Maximum der Zahlen ausgegeben werden.
17. Wandeln Sie das Beispiel “Zahlen sortieren” in ein `JApplet` um. Die Beschreibung der Eingaben soll über `JLabels` erfolgen, das Einlesen der Zahlen über `JTextFields` und die Ausgabe in einer `JTextArea`.
18. Schreiben Sie eine Applet-Version des Programmes ‘Zahlenraten’ mit folgender Funktionalität: Zunächst soll in einer nicht editierbaren `TextArea` eine kurze Erklärung des Spieles gegeben werden. Die Eingaben des Users sollen in ein `TextField` erfolgen. Falls die Eingabe höher ist als die vom System vorbereitete Zufallszahl, soll die Farbe des Applets auf rot wechseln und in einem nicht-editierbaren `TextField` sollen die Worte ‘zu hoch’ erscheinen. Andernfalls soll die Farbe auf ‘blau’ gesetzt werden und ‘zu niedrig’ in das Textfeld geschrieben werden. Falls der User die Zahl errät, soll im Textfeld ‘Richtig!!’ erscheinen und das Eingabe-Textfeld soll uneditierbar gesetzt werden. Durch Drücken eines Buttons soll ein neues Spiel begonnen werden können.
19. Schreiben Sie ein Applet, das es dem User erlaubt, durch Mausbewegung Rechtecke zu zeichnen: Die linke obere Ecke des Rechteckes soll an der Stelle sein, an der die linke Maustaste gedrückt wird. Dann soll der User durch Ziehen (Dragging) des Mausursors und anschließendes Loslassen die rechte untere Ecke des Rechteckes bestimmen können. In der Statuszeile soll die Fläche des jeweiligen Rechteckes angezeigt werden. Benutzen Sie zum Zeichnen der Rechtecke die `Graphics`-Methode `drawRect`, deren Parameterliste Sie am besten der Online-Dokumentation entnehmen.
20. Schreiben Sie ein `JApplet`, in dem mit Hilfe des geeigneten Layout-Managers 100 `JButtons` in einem 10x10-Feld angeordnet sind. Beim Klicken auf einen `JButton` soll dieser sowie zwei weitere zufällig ausgewählte Buttons vom Applet entfernt werden.
- 21* Schreiben Sie eine Applikation, mit der man geometrische Objekte in verschiedenen Farben in das Hauptfenster zeichnen kann. Das Hauptfenster in Ihrem Programm soll dabei eine Unterklasse von `JFrame` sein.

Die Anwendung soll eine Menüleiste mit den drei Menüpunkten *Programm*, *Farbe* und *Zeichnen* besitzen. Über das *Programm*-Menü soll man Informationen über das Programm (in einer `JOptionPane`) sehen (Autor, Beschreibung, ...). Außerdem gibt es hier einen Menüpunkt *Beenden*, über den man das Programm verläßt. Im Menü *Farbe* stehen einige Farben zur Auswahl. Im Menü *Zeichnen* findet man einige geometrische Objekte wie Rechteck oder Linie vor. Wenn man hier einen Menüpunkt auswählt, wird in der im *Farbe*-Menü gewählten Farbe dieses Objekt in einer zufälligen Größe an eine zufällige Position im Fenster gezeichnet.

22. Modifizieren Sie zwei Ihrer bisher geschriebenen Applikationen, die User-Eingaben erfordern so, daß diese mittels `try` und `catch` Exception-Handling verwenden. Definieren und verwenden Sie dazu eigene Unterklassen von `Exception`.
23. Schreiben Sie eine Applikation, die die gleichen Aufgaben wie das in der Vorlesung besprochene Beispiel `WriteFile` erfüllt. Die Ein- und Ausgabe soll jedoch über GUI-Komponenten erfolgen.
- 24* Schreiben Sie ein Applet, das eine gelbe Warnblinkanlage, wie Sie sie von einigen Kreuzungen her kennen, anzeigt. Das Applet soll die Möglichkeit bieten, das Blinken zu starten und stoppen. Implementieren Sie das Blinklicht als eigenen Thread.