



Dipl.-Inform. Gregor Fischer <fischer@informatik.uni-wuerzburg.de>

Einführung in die Informatik

für Hörer alle Fakultäten

(WS-03/04)

Übungsblatt 4

Ausgabe: Do. 2003-11-13

Abgabe: Do. 2003-11-20

Besprechung: Do. 2003-11-25/27

Vorbemerkung: Praktomat

Für die Programmieraufgaben aus dieser Sektion der Vorlesung müssen die Aufgaben über das automatische Abgabesystem **Praktomat** eingereicht werden.

Dieses wird eine Reihe von automatischen Tests mit dem von Ihnen eingereichten Programm durchführen, und Ihnen die Ergebnisse auch gleich mitteilen. Nicht alle durchgeführten Tests müssen bestanden werden. Wenn Ihr Programm einen Test besteht, heisst das allerdings auch nicht zwangsweise, dass es auch korrekt ist. Die Tests sind jedoch ein guter Hinweis darauf, wie Ihr Programm bei der abschliessenden manuellen Korrektur abschneiden wird. Sie können Ihr Programm bis zum Abgabetermin beliebig oft einreichen, prüfen lassen und bei Bedarf erneut nachbessern.

Den Praktomaten finden Sie unter der Adresse:

<https://jop.informatik.uni-wuerzburg.de/infohaf/praktomat.cgi>

Zur Benutzung des Praktomaten wurde Ihnen als Login ihre Matrikelnummer, und als Passwort ihr Geburtsdatum (im Format: JJMMTT) zugewiesen. **Bitte ändern Sie Ihr Passwort umgehend!**

Die Abgabe muss über den Praktomaten erfolgen, bis zu dem auf dem Aufgabenblatt angegebenen Abgabetermin, jeweils bis spätestens 23:59 Uhr.

Das Java-Tutorial, das Sie unterstützend verwenden können funktioniert mit den selben Logindaten, wie der Praktomat. Sie finden es unter:

<http://jop.informatik.uni-wuerzburg.de/tutorial/InfoHaF/>

Hinweis: Bei allen Aufgaben dieses Übungsblattes ist es nicht notwendig, dass Sie Eingaben auf ihre Korrektheit überprüfen. Sie können also z.B. davon ausgehen, dass wenn Zahlen in der Eingabe erwartet werden, auch solche eingegeben werden, oder dass die Koordinaten der Rechtecke auch sinnvoll angeordnet sind.

Aufgabe 4.1: Fläche (2 Punkte)

Schreiben Sie ein Programm mit der Hauptklasse `KreisFlaeche` welche eine Main-Methode (`public static void main(String[] args)`) enthält, das die Fläche eines Kreises berechnet.

Dazu wird dem Programm ein Kommandozeilenparameter mitgegeben, der den Radius `radius` des zu berechnenden Kreises angibt. Da Kommandozeilenparameter immer als Zeichenketten ausgefasst werden, müssen Sie diese zunächst noch, wie in der Vorlesung gezeigt, mit der [Simple-Bibliothek](#) in eine Zahl umwandeln. Damit berechnen Sie dann die Fläche des Kreises. Dabei können und sollen Sie die Konstante `Math.PI` verwenden.

Die Ausgabe des Programms soll ausschliesslich der Wert der berechneten Fläche sein.

Aufgabe 4.2: Umfang (2 Punkte)

Schreiben Sie ein Programm `RechteckUmfang`, das den Umfang eines Rechtecks berechnet. Dazu sollen zunächst in der `main`-Methode vom Benutzer die Koordinaten des Rechtecks eingelesen werden, und zwar in folgender Reihenfolge:

1. X-Koordinate der linken unteren Ecke
2. Y-Koordinate der linken unteren Ecke
3. X-Koordinate der rechten oberen Ecke
4. Y-Koordinate der rechten oberen Ecke

Dabei verwenden Sie wiederum die [Simple-Bibliothek](#), wie in der Vorlesung gezeigt.

Aus diesen Daten wird anschliessend die Umfang berechnet und dann nur das Ergebnis als Zahl ausgegeben.

Aufgabe 4.3: Überdeckung (6 Punkte)

Schreiben Sie ein Programm `Ueberdeckung`, das prüft, ob sich Kreise bzw. Rechtecke überdecken.

Dazu liest das Programm zunächst eine Zahl ein, die angibt, was geprüft werden soll:

- 1: Kreis mit Kreis.
- 2: Rechteck mit Rechteck.
- 3: Kreis mit Rechteck.

Für die Elemente werden dann entsprechen die erforderlichen Daten eingelesen.

Das sind für einen Kreis:

1. X-Koordinate des Mittelpunkts
2. Y-Koordinate des Mittelpunkts
3. Radius

für ein Rechteck:

1. X-Koordinate der linken unteren Ecke
2. Y-Koordinate der linken unteren Ecke
3. X-Koordinate der rechten oberen Ecke.
4. Y-Koordinate der rechten oberen Ecke.

Anschliessend soll das Programm feststellen, ob eine Überdeckung der Zeichenobjekte vorhanden ist, und entsprechen den Text `true` für "Überdeckung" bzw. `false` für "keine Überdeckung" ausgeben.

Hinweis: Machen Sie sich mit Hilfe von Skizzen klar, welche Anordnungen es gibt, und durch welche Bedingungen diese charakterisiert sind. Ihr Programm kann dann durch Abprüfen der Bedingungen feststellen ob eine Überdeckung vorliegt.