

Übungsblatt 05

E-Learning

Absolvieren Sie die Tests bis Di., 27.11., 23:59 Uhr.

Die Tests sind in der Stud.IP-Veranstaltung *Informatik I* unter *Lernmodule* hinterlegt.

Sie können einen Test **nur einmal durchlaufen**. Sobald Sie einen Test starten steht Ihnen nur eine **begrenzte Zeit** zu Verfügung, um den Test zu bearbeiten.

Alle Punkte, die Sie beim Test erreichen, werden ihnen angerechnet.

ILIAS – 24 Punkte

Anwendung von Wahrheitswerten

Absolvieren Sie den Test *Informatik I - ILIAS 05 Teil 1*.
(24 Punkte)

Hinweis

Wenn Sie den Test einmal vollständig durchlaufen haben kommen Sie auf die Seite *Testergebnisse*. Starten Sie den Test erneut aus Stud.IP, ist jetzt auch eine Schaltfläche *Testergebnisse anzeigen* vorhanden, die auf diese Seite führt.

Auf der Seite *Testergebnisse* können Sie sich unter *Übersicht der Testdurchläufe* zu jedem Testdurchlauf *Details anzeigen* lassen.

In der Auflistung der Aufgaben führt der Titel einer Aufgabe zu einer **Musterlösung** für die jeweilige Aufgabe.

ILIAS 4-Minuten-Aufgaben – 12 Punkte

Absolvieren sie die Tests *Informatik I - ILIAS 05 Teil 2*, *Teil 3* und *Teil 4*.
(12 Punkte)

LON-CAPA – 15 Punkte

Codierung DCF77

Informieren Sie sich **bevor Sie den Test starten** über den Zeitzeichensender DCF77, z.B. im zugehörigen Wikipedia-Artikel (<https://de.wikipedia.org/wiki/DCF77>).

Absolvieren Sie im Test *Informatik I - LON-CAPA* die *Übung 05*.
(15 Punkte)

Übung

Abgabe bis Di., 27.11., 18 Uhr.

Werfen Sie Ihre Lösung in den Zettelkästen Ihrer Gruppenübung. Für die Übungen im Nordbereich stehen die Zettelkästen im Sockelgeschoß (Ebene -1) **oder** auf dem Flur vor dem Seminarraum auf Ebene 0 des Instituts für Informatik.

Wenn Ihre Übung im Südbereich stattfindet, klären Sie mit Ihrem Tutor wo die Lösungen abzugeben sind.

Achten Sie darauf, dass Ihr **Name**, Ihre **Gruppe** und Ihre **Matrikelnummer** auf **jedem** Blatt stehen!

Falls Ihre Lösung mehrere Blätter umfasst, heften Sie diese bitte zusammen.

Aufgabe 1 – 14 Punkte

Polynomfunktion

Polynome sind Ausdrücke der Gestalt

$$p(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0$$

wobei die Koeffizienten des Polynoms $a_n, a_{n-1}, \dots, a_1, a_0$ beliebige Zahlen sind. Die Polynomfunktion $x \mapsto p(x)$ ordnet den Wert zu, der durch Einsetzen von x entsteht.

Sei

$$q(x) = a_n x^{n^2} + a_{n-1} x^{(n-1)^2} + \dots + a_2 x^{2^2} + a_1 x + a_0$$

ein spezielles Polynom mit quadratischen Exponenten und ganzzahligen Koeffizienten $a_n, \dots, a_0 \in \mathbb{Z}$.

Unmittelbar aus der Definition von q ergibt sich ein Algorithmus zur Berechnung des Funktionswertes der zugehörigen Polynomfunktion an der Stelle x .

Eingabe: Koeffizienten $a[n], a[n-1], \dots, a[0]$, Argument x

```
q = a[0];
for (i = 1; i <= n; i++) {
    potenz = 1;
    for (j = 1; j <= i*i; j++)
        potenz = potenz * x;
    q = a[i] * potenz + q;
}
return q;
```

1. Wie viele Multiplikationen und Additionen werden – in Abhängigkeit von n – mit diesem Algorithmus benötigt, um den Wert der Polynomfunktion an einer beliebigen Stelle x zu berechnen.
(2 Punkte)

Hinweis

Die Operationen, die von den **for**-Schleifen benutzt werden, um die Laufvariablen zu verändern, brauchen nicht berücksichtigt werden.

2. Wie kann man den Algorithmus verbessern?

- a) Beschreiben Sie eine Idee zur Verbesserung des Algorithmus, so dass die Anzahl der Multiplikationen deutlich reduziert wird.
(5 Punkte)

Hinweis

Maximales Ausklammern, Horner-Schema

- b) Geben Sie den verbesserten Algorithmus in Java/Pseudo-Code an.
(5 Punkte)
- c) Wieviele Multiplikationen und Additionen – in Abhängigkeit von n – benötigt der verbesserte Algorithmus?
(2 Punkte)

Praktische Übung

Abgabe der Prüfsumme bis Di., 27.11., 23:59 Uhr.

Testat von Mi., 04.12., 8-10 Uhr bis Mo., 10.12., 18-20 Uhr.

Hilfe zum Bearbeiten der praktischen Übungen können Sie grundsätzlich jeden Tag in den Rechnerübungen bekommen, insbesondere in den Rechnerübungen **Di., 27.11., 8-10 Uhr und 18-20 Uhr**, in denen **keine Testate** stattfinden.

Hinweise zu den praktischen Übungen, insbesondere zur **Abgabe der Prüfsumme** und zur **Durchführung der Testate**, sind in der Stud.IP-Veranstaltung *Informatik I* unter *Dateien*→*Übungsblätter* hinterlegt.

Aufgabe 1 – 10 Punkte

StdDraw

Laden Sie aus <https://www.stud.informatik.uni-goettingen.de/info1/java/> den Quellcode der Klassen `Triangle`, `Envelope` und `PlotFilter` herunter, die im Skript Kapitel 3.1.4 *Grafik mit der Klasse StdDraw* vorgestellt werden.

1. Kompilieren Sie den Quellcode und führen Sie die Applikationen aus.
2. Erläutern Sie die Funktionsweise der Klasse `Triangle` zeilenweise.
3. Verwenden Sie bei der Ausführung von `PlotFilter` die Daten aus der Datei `Deutschland.txt` als Eingabe. Die Datei `Deutschland.txt` ist in der Stud.IP-Veranstaltung *Informatik I* unter *Dateien*→*Übungsblätter*→*Daten* hinterlegt.

(10 Punkte)

Hinweise

- Die Bibliothek `stdlib.jar` finden Sie auf der Webseite *Standard Libraries* zum Buch *Introduction to Programming in Java* von Sedgewick/Wayne unter <https://introcs.cs.princeton.edu/java/stdlib/>.
- Wie Sie Bibliotheken beim Kompilieren und Ausführen zum Klassenpfad hinzufügen ist im Skript Kapitel 3.1.2 *Nutzerbibliotheken* beschrieben.

Graphik-Ausgabe

- Führen Sie das Java-Archiv `dragon.jar` mit gültigen Kommandozeilenparametern aus.
- Führen Sie das Java-Archiv `drawcurve.jar` aus und geben Sie folgende Daten ein.

- Führen Sie beide Java-Archive aus, leiten Sie dabei die Ausgabe von `dragon.jar` als Eingabe an `drawcurve.jar` weiter.

Hinweise

2. Schreiben Sie eine Applikation, die die Funktion von `drawcurve.jar` aus Aufgabenteil 1 nachbildet, d.h. gesteuert von einer Zeichenkette wird einen Polygonzugs gezeichnet.

F Bewege den Zeichenstift um die Länge 1 in die Richtung, in die der Zeichenstift orientiert ist und zeichne dabei eine Linie.

R Drehe den Zeichenstift um 90° nach rechts.

Aus der Zeichenkette **FLFLFRFLFLFRFRF** resultiert folgender Polygonzug.



Setzen Sie folgende Anforderungen um.

- Die Applikation liest die Anzahl der Zeichen und die Zeichen nacheinander von der Standardeingabe. Die eingelesene Zeichenkette wird in der Klasse als Feld von `char` verwaltet.

Beispiel für eine gültige Eingabe.

15 FLFLFRFLFLFRFRF

Hinweis. Als Vorbild können Sie den Code der Methode `readDouble1D` aus der Klasse `StdArrayIO` (Skript Kapitel 3.5.2 *Benutzerdefinierte Bibliotheken*) verwenden.

- Dem eigentlichen Zeichnen geht eine Simulation voran, die die Größe des Polygonzugs ermittelt, d.h. die maximale/minimale x-/y-Koordinaten bestimmt, und die Skalierung des angezeigten Koordinatensystems so festlegt, dass der komplette Polygonzug sichtbar ist.
- Der Polygonzug wird mit Hilfe der Methoden aus der im Skript vorgestellten Klasse `StdDraw` gezeichnet.
- Versehen Sie den Quellcode mit ausführlichen Kommentaren, die den Programmablauf erläutern.
- Testen Sie die Applikation mit dem Java-Archiv `dragon.jar` aus Aufgabenteil 1.

(20 Punkte)