

Zusatzaufgabe 1: Dreiecksfläche

Die Formel von Heron zur Berechnung der Fläche A eines Dreiecks aus seinen Seitenlängen a, b, c lautet:

$$A = \sqrt{s(s-a)(s-b)(s-c)}$$

wobei

$$s = \frac{a+b+c}{2}$$

1. Schreiben Sie eine Funktion, die diese Formel berechnet.
2. Finden Sie einen Weg, den Hilfsausdruck für s nur einmal hinschreiben zu müssen. Dazu brauchen Sie nur Konzepte, die bereits eingeführt wurden! (**Schwer**)
3. Finden Sie Testfälle.

Zusatzaufgabe 2: Tonhöhen (1/2)

Das MIDI-Format für Musikdaten stellt Tonhöhen durch ganze Zahlen im Bereich 0–127 dar, wobei die Werte 21–108 den Tasten eines handelsüblichen Klaviers entsprechen. Jede dieser Tonhöhen ist physikalisch durch eine Frequenz charakterisiert.

- Befolgen Sie **alle** in der Vorlesung gegebenen Regeln.
- Der Zusammenhang zwischen MIDI-Tonhöhe und Frequenz ist exponentiell, und standardmäßig so definiert, dass
 - der Tonhöhe 69 die Frequenz 440 Hz entspricht, und
 - sich alle 12 Tonhöhen die Frequenz verdoppelt (bei 81 also 880 Hz etc.)
- 1. Schreiben Sie eine Funktion, die MIDI-Tonhöhen in Frequenzen umrechnet.
- 2. Schreiben Sie eine Umkehrfunktion, die eine Frequenz auf die nächste ganzzahlige MIDI-Tonhöhe rundet.
- 3. Testen Sie auch, inwieweit die Hintereinanderanwendung beider Funktionen zum Eingabewert zurückkehrt. (**Schwer**)
- 4. Wie verhalten sich Ihre Funktionen, wenn der definierte Wertebereich verlassen wird? Erweitern Sie den Zweckbestimmungskommentar entsprechend.

Zusatzaufgabe 2: Tonhöhen (2/2)

- Schreiben Sie eine Funktion, die prüft, ob eine gegebene MIDI-Tonhöhe auf einem Audiogerät mit bekannter Minimal- und Maximalfrequenz ausgegeben werden kann.

Hinweise:

- Recherchieren Sie den Racket-typischen Gebrauch der Vergleichsoperatoren.
 - Zur Formulierung von Testfällen verwenden Sie die Funktion direkt (das Resultat ist bereits ein Wahrheitswert) ohne Vergleichsoperator.
- Nach dem Satz von Nyquist muss die Abtastrate eines digitalisierten Signals mindestens doppelt so hoch sein wie die darzustellende Signalfrequenz, damit keine störenden Artefakte auftreten.
 1. Schreiben Sie eine Funktion, die prüft, ob eine MIDI-Tonhöhe mit einer gegebenen Abtastrate so digitalisiert werden kann.
 2. Definieren Sie Konstanten für die folgenden üblichen Abtastraten:
Telefon (8 kHz), CD (44,1 kHz), Studio (96 kHz)
 3. Geben Sie einen Ausdruck an, der die höchste mit Studio-Rate digitalisierbare MIDI-Tonhöhe findet (diese kann außerhalb des definierten Bereichs liegen).

Zusatzaufgabe 3: Auswertung

Werten Sie Testfälle für schon gelöste Aufgaben (oder deren Musterlösungen) ebenfalls von Hand aus.

- **Hinweis:** Sie können zur Validierung die umgeformten Ausdrücke als Racket-Code untereinander schreiben. Jede Umformung sollte syntaktisch korrekt sein und dasselbe Ergebnis liefern.
- Überprüfen Sie Ihre Ergebnisse ebenfalls mit dem Stepper.

Zusatzaufgabe 4: Drachenfeuer

Drachen sind feuerspeiende Wesen, und daher sehr gefährlich. Im Alter nimmt einerseits ihre Feuerkraft zu, andererseits ihre Feuerbereitschaft ab, da ihnen mehrere Köpfe wachsen, die einander in Streitgespräche verwickeln.

Die Drachenforschung hat folgende empirische Schätzregel entwickelt:

- Die Feuerkraft eines Drachen beträgt bei Geburt 0,25 Hektogrill und steigt linear um 1 Hektogrill pro 666 Jahre Lebensalter.
- Der zweite Kopf wächst einem Drachen durchschnittlich nach Vollendung des 500. Lebensjahres, der dritte nach weiteren 1000 Lebensjahren, danach ein weiterer alle 2000 Jahre.
- Die Häufigkeit von Streitgesprächen ist proportional zur Anzahl der ungeordneten Paare (2-Kombinationen ohne Wiederholung) verschiedener Köpfe. Ein einköpfiger Drache hat einen Feuerbereitschafts-Faktor von 1. Jedes streitbare Paar reduziert die Feuerbereitschaft linear um 4 %.

Berechnen Sie die erwartete Gefährlichkeit eines Drachen als Produkt von Feuerkraft und -bereitschaft in Abhängigkeit von seinem Lebensalter.

Zusatzaufgabe 5: Pizza

Die Abmessung einer Pizza (Margherita) in der Speisekarte eines Lieferdienstes ist durch vier Größen charakterisiert:

- den Durchmesser, sowie
- die Höhen von Boden, Tomatensauce und Käse.

1. Definieren Sie eine Datenstruktur `pizza`
2. Berechnen Sie die Masse einer Pizza unter Annahme folgender Dichtewerte (in g / cm^3):

Boden	0,5
Tomatensauce	1,1
Käse	0,8

3. Schreiben Sie eine Funktion `mit-extra-kaese`, die einer gegebenen Pizza-Form 2 mm an Käsehöhe hinzufügt. (Alle anderen Kennzahlen bleiben gleich)
4. Ein anderer Mitarbeiter hat eine Datenstruktur für die vorhandenen Formate von Kartons verlangt. Diese soll einen Karton durch das Bruttovolumen und die Höhe charakterisieren. Schreiben Sie diese, sowie eine Funktion `passt-in?`, die ermittelt, ob eine gegebene Pizza in einen gegebenen Karton passt. Dazu muss die Dicke des Kartons (1,5 mm) in allen Dimensionen berücksichtigt werden.

Zusatzaufgabe 6: Kleidung (1/2)

Ein Smart-Home-Kleiderschrank soll die intelligente Verwaltung von Kleidungsstücken übernehmen. Die Software V1.0 soll Hosen, Gürtel und Socken unterstützen.

- Eine Hose ist charakterisiert durch Farbe (als Nummer im ► RAL-Katalog), sowie Bundweite und Länge (beide in Zoll).
 - Socken sind charakterisiert durch Farbe und Schuhgröße (EU).
 - Gürtel sind charakterisiert durch Farbe und Länge (in cm).
1. Definieren Sie je eine Datenstruktur für jede der drei Arten von Kleidungsstücken, sowie einen allumfassenden gemischten Datentyp.
 2. Machen Sie im Folgenden bewusst von Regeln 1–5 Gebrauch. Dokumentieren und testen Sie alle Definitionen unabhängig und gründlich.
 - **Hinweis:** Zahlen treten hier mit vielen unterschiedlichen Bedeutungen auf — nur ausführliche Dokumentation schützt zuverlässig vor Verwechslung!
 3. Definieren Sie eine Funktion, die die Farbe eines beliebigen Kleidungsstücks ermittelt.

Zusatzaufgabe 6: Kleidung (2/2)

4. Schreiben Sie eine Funktion, die ermittelt, ob zwei beliebige Kleidungsstücke zusammenpassen. Dabei gelten folgende Regeln:
- Kleidungsstücke gleicher Art (z. B. Socken zu Socken) passen nie;
 - Gürtel passen zu Hosen, wenn Sie ≥ 15 cm länger sind als die Hose weit.
 - Socken passen zu Hosen, wenn die Sockengröße den geometrischen Mittelwert von Hosenlänge und -weite (in Zoll) um 15–40 % überschreitet.
 - Socken passen zu Gürteln, wenn sie die gleiche Farbe haben.

Das Ergebnis darf nicht von der Reihenfolge der Kleidungsstücke abhängen!

5. Schreiben Sie eine Funktion, die zu einem beliebigen Kleidungsstück einen Vorschlag für ein (nach den obigen Kriterien) passendes zweites Stück berechnet. Wie gehen Sie mit der Wahlfreiheit um?
6. Fügen Sie für Software V2.0 Schuhe hinzu. Diese sollen ebenso wie Socken charakterisiert werden, und die gleichen Passkriterien erfüllen. (**Schwer**) Welche Arbeitsschritte müssen Sie für diese Erweiterung ausführen, damit alle bisherigen Definitionen weiter funktionieren?

Zusatzaufgabe 7: Arithmetik (1/2)

Stellt man einem arithmetischen Ausdruck in Racket (bestehend aus Zahlen, Operatoren und Klammern) einen `'`-Operator voran, erhält man einen **arithmetischen S-Ausdruck**. Beispiele:

```
'3  
'(+ 1 2 3)  
'(* 5 (+ 4 2))
```

1. Schreiben Sie eine Funktion `evaluate`, die einen solchen Ausdruck analysiert und zu einer Zahl auswertet. Dabei müssen zunächst nur Zahlen und die Operatoren `+` und `*` erkannt werden, diese sollen aber mit beliebig vielen Operanden funktionieren.
 - Hinweis: Schreiben Sie eine passende Hilfsfunktion für jeden Operator, oder verwenden Sie Lösungen von Aufgabe 17 wieder.
 - Erinnern Sie sich außerdem an die allgemeinen Auswertungsregeln.

Zusatzaufgabe 7: Arithmetik (2/2)

2. Erweitern Sie ihre Funktion um einen Operator `^` (der so in Racket nicht vordefiniert ist). Dabei soll `'(^ x y1 ... yn)` den Wert $(x^{y_1} \dots)^{y_n} = x^{y_1 \dots y_n}$ ergeben.
 - Hinweis: Sie können die Racket-Funktion `expt` verwenden, die aber nur mit genau zwei Operanden funktioniert.
 3. Erweitern Sie ihre Funktion so, dass die Konstante `pi` verwendet werden kann.
 4. Modifizieren Sie die Funktion so, dass sowohl die symbolische Konstante `pi` als auch jeder Ausdruck, der denselben Wert ergibt, auf `22/7` gerundet wird. (**Schwer**)
- Sie finden Testfälle für jeden Schritt und eine Schablone als Vorgabe.

Zusatzaufgabe 8: Sortieren

Auf Basis der in Aufgabe 18 entwickelten Funktion `merge` kann ein Algorithmus `merge-sort` zum Sortieren einer Liste von Zahlen entwickelt werden:

- Eine Liste mit höchstens einem Element ist bereits sortiert.
- Eine Liste mit zwei oder mehr Elementen kann in zwei (möglichst gleichlange) Teillisten aufgeteilt werden. Diese werden (unabhängig voneinander) rekursiv sortiert, und die Ergebnisse mit `merge` zusammengefügt.
 - Finden Sie ein möglichst einfaches Verfahren zum Teilen einer Liste.

Zusatzaufgabe 9: Transfer

1. Fügen Sie die in Aufgabe 20 programmierte Funktion `diff` in die Lösung von Zusatzaufgabe 7 ein.
 - Sie müssen dazu die Codierung der Operatoren anpassen.
 - Übertragen Sie zuerst die Testfälle, dann die Implementierung.
2. Erweitern Sie `diff` um die Behandlung des Operators `^`.
3. Erweitern Sie umgekehrt die Verwendung von Variablen und `'ERROR` auf `evaluate`. (**Schwer**)