



## Übungsblatt 0

Themen: Einführung, Informationen und Einstieg Racket

Relevante Folien: Orgafolien, Funktionale Abstraktion

Abgabe der Hausübung: 26.10.2018 bis 23:55 Uhr

**Wichtiger Hinweis:** Es ist immens wichtig, dass Sie sich intensiv und gewissenhaft mit den hier vorgestellten Themen beschäftigen, da sie die Grundlage für die Hausübungsabgaben in diesem Semester darstellen. Auf dieser und der nächsten Seite erhalten Sie einige allgemeine Informationen, diese gelten für alle Hausübungen und wir setzen diese Informationen als gegeben voraus.

### Aufbau der Übungsblätter

Die Übungsblätter gliedern sich immer in zwei Teile - die vorbereitenden Übungen **V** und die Hausübungen **H**. In den vorbereitenden Übungen sollen die Konzepte der Vorlesung eingeübt werden. Die Aufgaben sind meist kürzer und sollen Sie praktisch an die Thematik heranzuführen. Außerdem bilden sie die Grundlage für die Hausübungen. Die Schwierigkeiten der einzelnen Übungen sind durch Sterne dargestellt (mindestens 0 und maximal 3), je mehr Sterne desto komplexer eine Aufgabe. Für ein sicheres Bestehen (das heißt 50% der Punkte) sollten Sie zumindest die Zwei-Sterne Aufgaben gut meistern können und die Drei-Sterne Aufgaben spätestens nach kleineren Hilfestellungen. Die Hausübungen sind meist etwas schwieriger als die vorbereitenden Übungen und werden am Ende bewertet. Zu den vorbereitenden Übungen wird es **keinen Lösungsvorschlag** geben! Die Lösungen der Hausübungen stellen wir Ihnen nach der Abgabefrist als Code auf moodle bereit.

Es gibt außerdem zu jedem Übungsblatt eine korrespondierende Fingerübung! Für nähere Informationen dazu, werfen Sie einen Blick auf Fingerübung 0.

### Programmieren mit Stift und Papier

Bei Fragen zu den Übungen bieten wir Ihnen unterschiedlichste Hilfsangebote. Beachten Sie jedoch, dass Sie in der Abschlussklausur keine Hilfsmittel zur Verfügung haben werden. Üben Sie also schon zu Beginn auch ohne Entwicklungsumgebung und nur mit Stift auf einem Blatt Papier zu programmieren. Sie lernen das Ganze aber nur wenn Sie sich selbst intensiv mit dem Thema auseinandersetzen, nutzen Sie Hilfe also nur wenn es gar nicht anders geht. Nur aus eigenen Fehlern lernen Sie richtig!

## Hilfsangebote

- Treffpunkt/Hörsaalübung: Der Treffpunkt findet jeweils zu angekündigten Daten nach der Dienstagsvorlesung im Audimax statt. Hier werden teilweise Übungen besprochen und Hilfestellungen zur Bearbeitung gegeben.
- Sprechstunden: Auf der Kursseite finden Sie eine Übersicht über die Sprechstunden, die angeboten werden. Diese können Sie aufsuchen und den Tutorinnen und Tutoren gezielt Fragen zu den Inhalten stellen.
- Forum: In moodle sind Foren für die Übungsblätter eingerichtet worden. Stellen Sie dort Ihre Fragen mit einem möglichst aussagekräftigen Titel, so können auch alle anderen Kursteilnehmer maximal davon profitieren.

## Abgabe der Hausübungen

Zur Hausübungsabgabe nutzen wir die entsprechenden Module in moodle. Die Abgabefristen sind auf jedem der Blätter vermerkt, sie liegen jedoch immer Freitagabends um 23:55 Uhr Serverzeit. **Es werden nur Bearbeitungen bewertet, die auf moodle hochgeladen wurden** (keine Repository-Links, keine Dropbox, keine E-Mails etc.). Sollte Ihnen beim Upload der Status “verspätet“ angezeigt werden ist dies kein Problem. Solange Sie etwas auf moodle hochladen können, akzeptieren wir es noch. Halten Sie sich bei der Abgabe der Dateien an die folgenden von uns gegebenen Konventionen! Die Konventionen für Racket-Dateien finden Sie in dieser Hausübung, die für Java-Dateien in Hausübung 3. Bei allen Abgaben handelt es sich um **Einzelabgaben**! Auch wenn sie zusammenarbeiten, müssen sie unterschiedliche und eigen bearbeitete Lösungen abgeben!

## Plagiarismus

Der Fachbereich Informatik misst der Einhaltung der Grundregeln der wissenschaftlichen Ethik großen Wert bei. Zu diesen gehört auch die strikte Verfolgung von Plagiarismus. Sollten wir ein Plagiat entdecken, wird dies entsprechend von uns geahndet.

[https://www.informatik.tu-darmstadt.de/studium\\_fb20/im\\_studium/studienbuero/plagiarismus/index.de.jsp](https://www.informatik.tu-darmstadt.de/studium_fb20/im_studium/studienbuero/plagiarismus/index.de.jsp)

Jeder der nachfolgenden Punkte ist vor allem für die FOP relevant, da sie Formen von Plagiarismus darstellen und damit verboten sind:

- Das Abschreiben von Lösungen von Hausübungen und Übernehmen von Lösungs-Code zu Programmieraufgaben.
- Das Übernehmen von fremderdachten Lösungsansätzen ohne korrekte Zitierung.
- Das Ausgeben eigener Hausübungslösungen vor dem Abgabetermin an andere.

Beachten Sie vor allem auch den letzten Punkt. Wenn Sie Ihre Lösung an jemand anderen weitergeben, so machen Sie sich selbst eines Plagiats schuldig und müssen ebenfalls mit Maßnahmen rechnen.

## V Vorbereitende Übungen

### V1 DrRacket installieren



Installieren Sie zu Beginn DrRacket von folgendem Link auf Ihrem Rechner:

<http://racket-lang.org/download/>

Alternativ können Sie DrRacket auch auf den Poolrechnern in S202 nutzen. In Abbildung 1 sehen Sie, wie die Oberfläche dann aussieht.

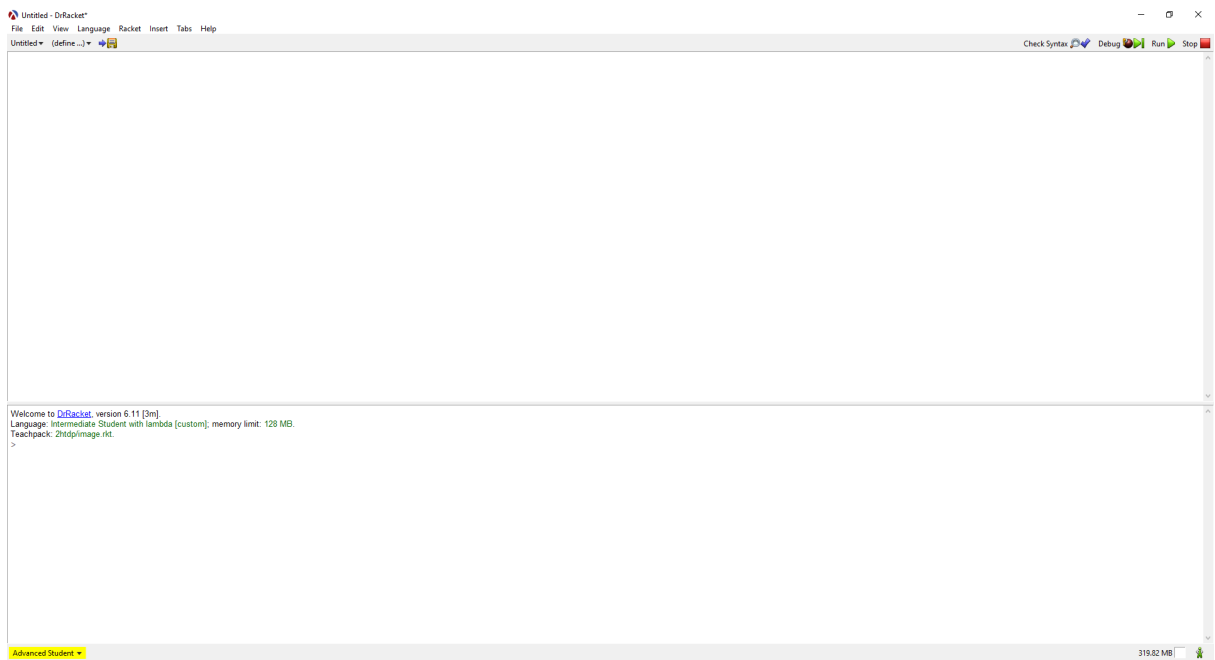


Abbildung 1: Oberfläche von DrRacket

Vergewissern Sie sich, dass links unten (im Screenshot gelb hinterlegt) auch *Advanced Student* steht. Andernfalls klicken Sie darauf und stellen es um.

Für erste Schritte in DrRacket, beachten Sie die dazugehörige Präsenzübung. Die Möglichkeiten der Entwicklungsumgebung wird auch ein Thema des ersten Treffpunkts sein.

## V2 Arithmetik



**Teil A:** Ergänzen Sie folgende Ausdrücke durch Klammern, so dass jeweils ein gültiger Racket-Ausdruck entsteht. Werten Sie anschließend diese Ausdrücke aus. Gehen Sie für diese Aufgabe davon aus, dass alle Operatoren genau zwei Operanden erwarten.

1. `+ + + 7 3 4 * 2 3`
2. `* + 1 9 - 4 + 1 1`
3. `* - 0 * 2 3 - / 3 3 - 5 6`
4. `+ * 5 5 - - * 3 3 - * 4 4 4 4`

**Teil B:** Übersetzen Sie die folgenden mathematischen Ausdrücke in äquivalente Racket-Ausdrücke. Die Ausdrücke sollen nicht vereinfacht (gekürzt oder berechnet) werden.  $4^3$  dürfen Sie durch eine entsprechende Multiplikation ersetzen. Um Ihre Eingaben zu überprüfen, haben wir Ihnen rechts das zu erwartende Ergebnis angegeben.

1.  $\frac{13-3}{6 \cdot 3} \cdot (100 - 32)$  [= 37.7]
2.  $(5 \bmod 3)/3.141 - 2$  [ $\approx -1.3632$ ]
3.  $\frac{4^3}{8} + (4 - 12)$  [= 0]

## V3 Steuern



Zu jeder von Ihnen definierten Funktion wird ein Vertrag und mindestens 3 nichttriviale Tests erwartet. Mit nichttrivialen Tests sind insbesondere solche gemeint, die Randfälle abdecken, andererseits repräsentative “normale” Fälle! Nutzen Sie dafür `check-expect` und `check-within`. Vergleichen Sie dazu auch das Beispiel und die Erläuterung im Hausübungsabschnitt und in der Vorlesung. In dieser Aufgabe steht es nochmals explizit dabei, ab dem nächsten Blatt nicht mehr.

Die Steuern in Land X ergeben sich aus dem Einkommen multipliziert mit dem Steuersatz. Der Steuersatz ist wiederum 0,5% je tausend Euro Einkommen (abgerundet). So würde ein Einkommen von 40.000 Euro einem Steuersatz von  $0,5\% \cdot 40 = 20\%$  und einer zu zahlenden Steuer von 8.000 Euro entsprechen, ein Einkommen von 23.700 Euro einem Steuersatz von 11,5% etc. Bei Einkommen größer als 200.000 Euro bleibt der Steuersatz bei 100%.

1. Definieren Sie eine Funktion `get-taxrate`, die ein Einkommen als Parameter erhält und den Steuersatz (in Prozent, d.h. 40 für 40%) zurückgibt. Nutzen Sie dazu die Funktion `floor`, die eine gegebene Zahl abrundet.
2. Definieren Sie eine Funktion `get-income`, die ein Einkommen übergeben bekommt und das Einkommen nach Abzug der Steuer zurückgibt.

## H Nullte Hausübung

Gesamt 1 Punkt

### *Abgabe von Racket Hausübungen*

Diese nullte Hausübung soll Sie mit den Abgabeformalitäten für die Racket Hausübungen vertraut machen. Für alle nachfolgenden Abgaben (in Racket) gilt:

Die hier vorgestellten Regeln müssen von Ihnen eingehalten werden! Dazu gehören zum einen die Vorgaben zum korrekten Abgabeprozedere, als auch die inhaltlichen Vorgaben von uns.

Für alle Hausübungen gibt es eine Vorlagendatei von uns in moodle. Nutzen Sie immer diese Vorlage und schreiben Sie keine eigene! Speichern Sie zunächst die heruntergeladene Vorlage an einem Ort Ihrer Wahl. Benennen Sie die Datei dann gleich korrekt um. Die Datei haben Sie dabei folgendermaßen zu benennen: `Hnr_In_fn`. Dabei ist `nr` durch die aktuelle Übungsnummer zu ersetzen (bei einstelligen Zahlen ergänzen Sie eine führende 0), `In` durch Ihren Nachnamen und `fn` durch ihren Vornamen. Max Mustermann gibt die Lösungsdatei für diese Übung also als **H00\_Mustermann\_Max.rkt** in moodle ab. Importieren Sie diese dann in DrRacket über *File* dann *Open* oder über den Shortcut `Strg+O`. Dort finden Sie bereits die Funktionen, die Sie in den Hausübungen zu definieren haben. Sie müssen die Funktionen also nur noch ergänzen. Möchten Sie Hilfsprozeduren definieren, so können Sie dies in der Vorlage einfach tun. Sie sind in der Anzahl der Hilfsfunktionen nicht beschränkt, achten Sie aber auf eine sinnvolle Zerlegung der Teilprobleme und lagern Sie nur dort aus, wo es auch sinnvoll ist.

Für **alle von Ihnen geschriebenen Funktionen** (egal ob vorgegebene Hauptfunktion oder neugeschriebene Hilfsfunktion) gilt:

Sie müssen Vertrag (Typ, Rückgabe und evtl. Precondition) und drei nichttriviale Tests für alle Funktionen angeben! Bei Lambda-Ausdrücken dürfen Sie auf die Tests verzichten, Vertrag ist aber auch hier verpflichtend. Orientieren Sie sich dabei an folgendem Beispiel, dass Ihnen schon aus der Vorlesung vertraut sein sollte (deutsch oder englisch ist egal):

```
;; Type: number number -> number
;; Returns: Euclidean norm of the vector (x,y)
(define (euclid2 x y)
  (sqrt (+ (* x x) (* y y))))
;; Tests
(check-expect (euclid2 3 4) 5)
(check-within (euclid2 2 2) 2.8284 0.001)
(check-within (euclid2 5.3 4.22) 6.774 0.001)
```

**Achtung:** Mit nichttrivialen Tests sind insbesondere solche gemeint, die Randfälle abdecken, andererseits repräsentative “normale“ Fälle! Nutzen Sie dafür `check-expect` und `check-within`. Testen Sie also nicht nur mit trivialen Eingabedaten (im Beispiel oben bspws. Kombinationen aus 0 und 1).

Haben Sie die Hausübung bearbeitet und möchten Sie abgeben, so speichern Sie Ihre Ergebnisse und geben Sie diese als Datei mit `.rkt` Endung ab!

Wenn Sie eine der auf dem Blatt beschriebenen Anforderungen zur Abgabe und Dokumentation verletzen, so werden wir Ihnen Punkte von Ihrer Gesamtpunktzahl abziehen! Vermeiden Sie dies also, um nicht unnötig Punkte zu verlieren.

Laden Sie Ihre Datei dann rechtzeitig im Abgabemodul in moodle hoch. Sie können Ihre Abgabe innerhalb der Bearbeitungsfrist beliebig oft hochladen, die zuletzt hochgeladene Version wird bewertet.

**Verbindliche Anforderung:** Für alle Racket-Hausübungen gilt, dass Zuweisung mittels `set!` oder ähnliches nicht erlaubt ist! Sollten Sie durch eigene Recherche auf dieses Konstrukt stoßen dürfen Sie es nicht verwenden, Sie müssen die Hausübungen mittels normaler Rekursion, wie in der Vorlesung vorgestellt, lösen.

## H1 Richtigen Dateinamen bilden

1 Punkt

Laden Sie sich die Vorlage dieser Hausübung herunter.

Ergänzen Sie dort die Funktion `generate-filename`. Diese bekommt die Zahl als `nr`, welche die aktuelle Blattnummer angibt, und zwei Strings `ln` und `fn`, die den Vor- beziehungsweise Nachnamen angeben. Die Funktion gibt die korrekte Dateibenennung für die entsprechende Hausübung zurück.

Beachten Sie nochmals die Erläuterungen zur korrekten Benennung vor der Aufgabe.

Vergessen Sie nicht Vertrag und mindestens drei Tests!

Diese Aufgabe ist sehr einfach, sie dient hauptsächlich dazu, Sie mit den Abgabeformalitäten vertraut zu machen. Sie erhalten diesen Punkt deswegen auch nur, wenn wirklich **alles** eingehalten und beachtet wurde. Verschenken Sie diesen Punkt also nicht, so einfach können Sie dieses Semester keinen mehr ergattern.

*Hinweise:*

- Vergessen Sie die Dateiendung `".rkt"` nicht!
- Mittels `string-append` können Sie beliebig viele Strings zusammenfügen.  
Beispiel: `(string-append "Hallo " "Welt" "!")` gibt `"Hallo Welt!"` zurück.
- Mittels `number->string` können Sie eine Zahl in einen String konvertieren.  
Beispiel: `(number->string 42)` gibt `"42"` zurück.