

## Fragen, Antworten, Kommentare zur aktuellen Vorlesung

Hinweis: Die Diskussion einer Beispiellösung zur Aufgabe 22 mit equals(.) für Linie finden Sie unter <https://youtu.be/2gpR9bMwSLY> (30:07).

Kommentar zur Aufgabe mit dem equals für die Linie:

Leider hat nicht jeder genau verstanden, dass man die Methode nicht alleine frei neu schreiben sollte. In der Vorlesung wurde ein klarer (Standard-)Weg vorgegeben, wie ein equals() entstehen soll. Zuerst bei jeder einzelnen Objektvariable, die als Typ eine echte Klasse hat, schrittweise einen Vergleich machen und nach Unterschieden suchen. Am Ende werden dann die Objektvariablen, die einen elementaren Typen (int, boolean, double, ...) direkt mit == auf Gleichheit geprüft [gibt es bei Linie nicht]. Natürlich dürfen erfahrenere Leute mit der Lösung experimentieren und sie umgestalten, da es durchaus sinnvolle Lösungsvarianten gibt. Dies gilt für Personen, die mit der Programmierung anfangen nur, wenn Sie die Standard-Lösung umgesetzt und genauso wichtig, das Konzept verstanden haben. Zum Verständnis des Konzepts tragen Aktivitätsdiagramme bei.

Frage: Ist der Nutzungsdialog so ok?

```
void dialog() {
    this.io.ausgeben("(0) Ende (1) Kreis (2) Dreieck: ");
    int eingabe = this.io.leseInteger();
    if (eingabe == 1){
        this.kreisEingeben();
    }
    if (eingabe == 2){
        this.dreieckEingeben();
    }
    if (eingabe != 0){
        dialog();
    }
}
```

Nein. Am Ende ruft die Methode sich selbst auf, was im Beispiel kein Problem ist, generell aber dazu führen kann, dass innerhalb einer Methode die gleiche Methode beliebig oft aufgerufen wird. Erinnern Sie sich an den Debugger, da werden die Aufrufe nicht abgeschlossener Methoden auf einen Stapel gelegt, der nur endlich viel Speicher hat und es so irgendwann zu einem Speicherüberlauf führen kann. Der generelle Ansatz, dass eine Methode sich selbst direkt oder über andere Methoden aufruft wird Rekursion genannt und kann für bestimmte Aufgaben, aber nicht für einen Dialog, sinnvoll sein. Die sinnvolle Nutzung von Rekursion ist später ein Thema im Studium und spielt in Programmierung 1 keine Rolle.

Frage: Bei Aufgabe 25e, was ist genau mit Schnitt gemeint?

Es soll der Durchschnitt *aller* bisher eingegebenen Zahlen berechnet werden. Dazu wird keine Liste benötigt, in der alle Zahlen drinstehen. Überlegen Sie welche Informationen Sie brauchen, um einen Durchschnitt zu berechnen. (Mit einer ArrayList geht es natürlich auch, sollte aber hier ohne gehen.)

Frage: Ich habe die Aufgaben 6 und 13 zu den Ausdrücken mal wiederholt. Mir ist noch unklar ob Zuweisungen und Methodenaufrufe Ausdrücke sind.

Zuweisungen sind als Befehle kein Ausdruck, da sie keinen Wert zurück liefern.

Googelt man sich da rein, ist eine reine Zuweisung schon ein Ausdruck. Das haben wir in der Vorlesung nicht behandelt und spielt in der Klausur keine Rolle:

Zuweisung als Befehl: `c=a+1;`

reine Zuweisung: `c=a+1`

Also ist folgender Befehl syntaktisch korrekt:

`a=b=c+1;`

(der Wert vom Ausdruck `c+1` wird der Variablen `b` zugewiesen, dann wird der Wert von `b` `a` zugewiesen) Wird seltenst benutzt, da es immer irritiert, findet man manchmal in einer Variante bei Schleifen:

```
int x;
while ((x = o.fkt(y)) != 4){
    // hier kann aktueller Wert von x genutzt werden
    erst fkt(y) ausrechnen, den Wert x zuweisen, dann mit 4 vergleichen.
```

Ein Methodenaufruf ist ein Ausdruck, wenn das Ergebnis nicht void ist, also ist `d.mathA(c)` in der Beispielaufgabe ein Ausdruck.

Nebenbei könnte bei A 13 jetzt aufgefallen sein, dass `methA(c);` durchaus ein sinnvoller Befehl sein könnte, wenn `methA(.)` eine Klassenmethode wäre, was hier nicht der Fall ist.

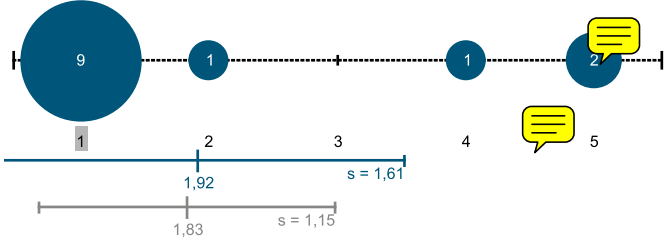
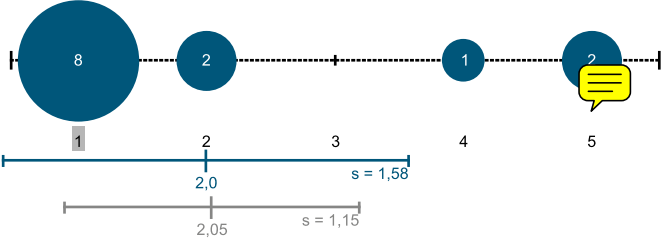
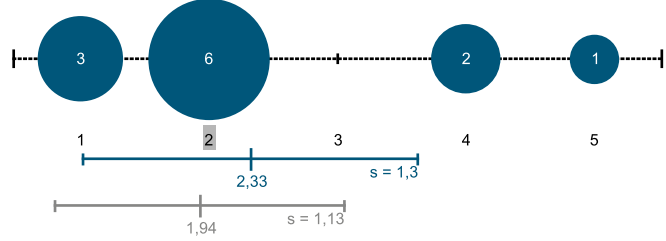
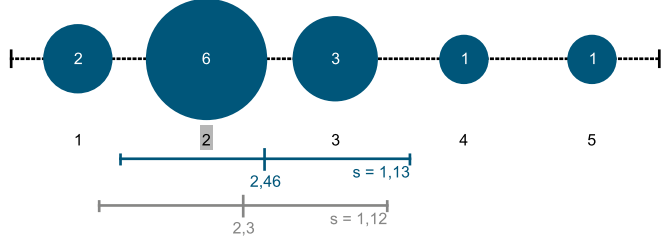
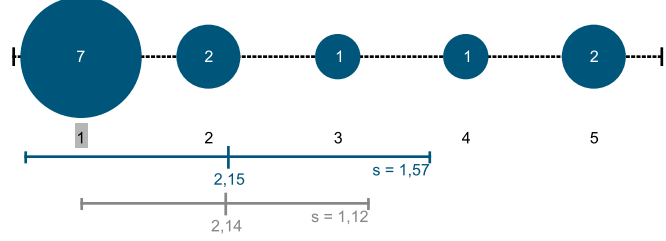
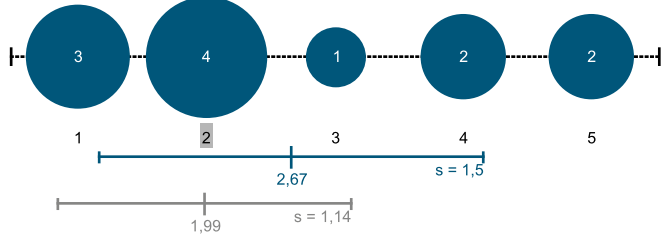
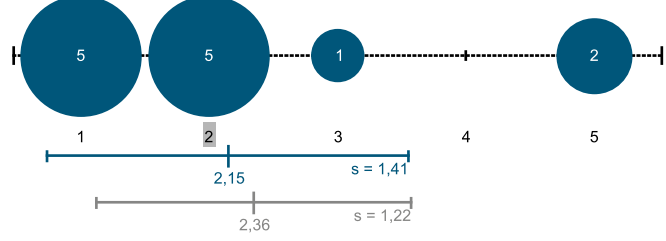
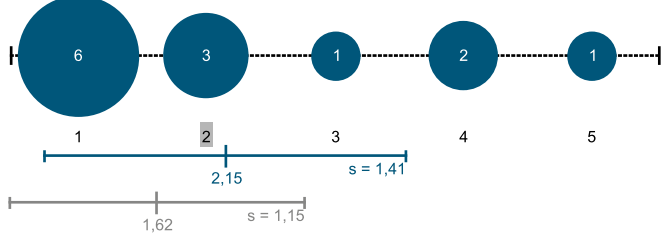
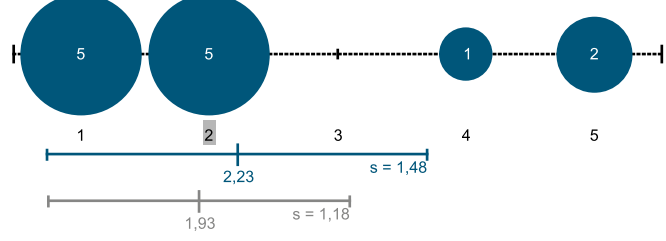
Beim Blick auf die Probeklausur fällt auf, dass genau eine solche Aufgabe nicht drin vorkommt, das genau wegen der hier genannten Spezialfälle.

Im Anhang finden Sie die Lehrevaluation mit einzelnen Kommentaren von mir. Bitte nehmen Sie auch an der Evaluation am Ende des Semesters teil, unabhängig ob Sie diesmal teilgenommen haben. Rückfragen zu den Themen sind jederzeit möglich.

## Auswertung zur Veranstaltung Programmierung 1 (I) (Vorlesung) - TI1/BB11

Liebe Lehrende,  
 anbei erhalten Sie die Ergebnisse der Evaluation Ihrer Lehrveranstaltung.  
 Zu dieser Veranstaltung wurden 13 Bewertungen (bei 40 Teilnehmenden) abgegeben. Dies entspricht einer Rücklaufquote von 33%.  
 Erläuterungen zu den Diagrammen befinden sich am Ende dieses Dokuments.  
 Mit freundlichen Grüßen,  
 Das Evaluationsteam

### Lehrprozess

 <p>Die Lehrveranstaltung verläuft nach einer strukturierten Gliederung</p>	 <p>Die Lernziele der Lehrveranstaltung werden klar definiert</p>
 <p>In der Lehrveranstaltung werden aktuelle Inhalte vermittelt</p>	 <p>Die Inhalte der Lehrveranstaltung sind gut mit anderen Lehrveranstaltungen des Studiums verknüpft</p>
 <p>In der Lehrveranstaltung werden oft Anregungen zur selbständigen Arbeit gegeben</p>	 <p>In der Lehrveranstaltung besteht häufig die Möglichkeit einer aktiven Teilnahme (u.a. an Diskussionen, Übungen, etc.)</p>
 <p>Die Darstellung der Lehrinhalte der Lehrveranstaltung weckt mein Interesse</p>	 <p>In der Lehrveranstaltung wird auf Fragen eingegangen</p>
 <p>Die Lehrveranstaltung wird durch eine gute Betreuung begleitet</p>	

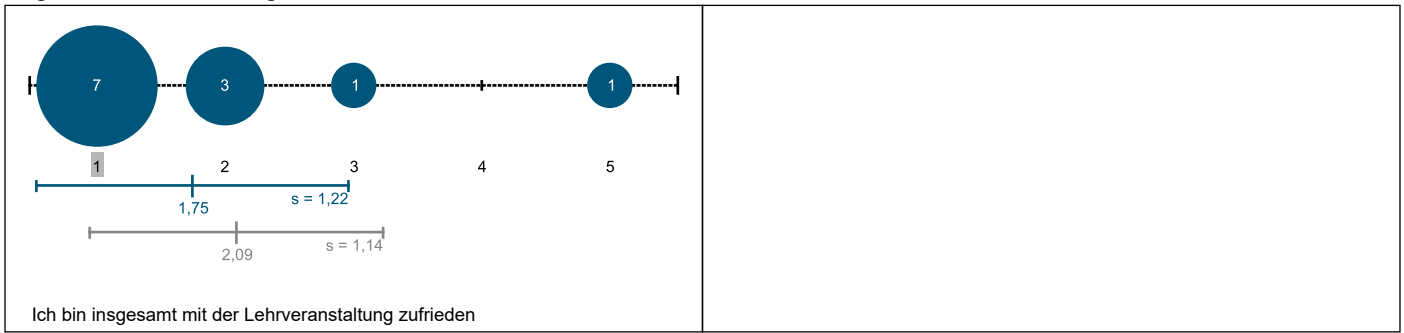
Lernprozess

<p>Ich bereite mich regelmäßig auf die Lehrveranstaltung vor</p>	<p>Ich bearbeite die der Lehrveranstaltung zugrundeliegende Literatur gründlich</p>
<p>Das Bearbeiten der Fachtexte hilft mir beim besseren Verständnis der Lehrinhalte</p>	<p>Ich habe aktiv an verschiedenen Übungselementen der Lehrveranstaltung teilgenommen, wie z.B. an Kleingruppenarbeiten, Diskussionsrunden, Fallstudien oder sonstigen Hörsaalübungen</p>
<p>Die Übungen haben mir beim Verständnis des Stoffes geholfen</p>	<p>Ich verfüge über ein besseres Verständnis des Themas als vor der Lehrveranstaltung</p>
<p>Die Lehrveranstaltung fördert mein Interesse am Studium</p>	<p>Mein durchschnittlicher Arbeitsaufwand für die Lehrveranstaltung, inkl. deren Besuch (in Zeitstunden pro Woche)</p>

Praxisbezug und Praxisrelevanz

<p>Der Bezug zwischen Theorie und Praxis wird hergestellt</p>	<p>Durch die Lehrveranstaltung werden potentielle Berufsfelder klarer</p>
---	---

## Allgemeineinschätzung



### Anregungen, Lob und Kritik (IN DRUCKSCHRIFT)

- Bin mit allen Lehrenden (sowohl im Praktikum als auch in den Vorlesungen) super zufrieden. Besser könnte es nicht sein!
- Herr Kleuker ist für mich der der beste Professor, die Inhalte sind super verständlich und auch sehr interessant
- Ich finde die Vorlesung sehr gut verständlich und das Video-Format ermöglicht es, das Tempo individuell anzupassen sowie die Vorlesung gut in den Alltag zu integrieren.
- Ich finde sehr gut, dass die Veranstaltung online stattfindet. Gerade im fach programmieren kann man so sehr gut und in seinem Tempo lernen. Bei anderen Lehrveranstaltungen würde ich die Vorlesung ungern online haben, hier hingegen ist das super. Dadurch das die Vorlesungen online sind ist das Studium wesentlich flexibler, das würde ich in diesem fach auf jeden Fall beibehalten.
- sehr gut



## Legende

