



Techn. Fakultät - Erwin-Rommel-Str. 60 - 91058 Erlangen

Dipl.-Inf. Jens Schedel  
(PERSÖNLICH)

## SS 13: Auswertung für Übungen zu Systemprogrammierung 1

Sehr geehrter Herr Dipl.-Inf. Schedel,

Sie erhalten hier die Ergebnisse der automatisierten Auswertung der Lehrveranstaltungsevaluation im SS 13 zu Ihrer Umfrage vom Typ "Übung":

- Übungen zu Systemprogrammierung 1 -

Es wurde hierbei der Fragebogen - u\_s13 - verwendet, es wurden 44 Fragebögen von Studierenden ausgefüllt.

Die Note 1 kennzeichnet hierbei eine maximale Güte, die Note 5 eine minimale Güte für die einzelnen Fragen bzw. Mittelwerte.

Auf der nächsten Seite zeigt der zuerst angegebene "Globalindikator" Ihre persönliche Durchschnittsnote über alle Kapitel-Indikatoren, deren Noten danach folgen.

Der Kapitel-Indikator für "Globalfragen für alle LV-Typen" ist trotz der Prozentangaben bei den Einzelfragen momentan noch ungewichtet, eine E-Mail mit dem daraus berechneten Lehrqualitätsindex (LQI) wird noch nachgeliefert.

Für die Ergebnisse aller Einzelfragen werden je nach Fragen-Typ die Anzahl und Verteilung der Antworten, Mittelwert und Standardabweichung aufgelistet.  
Die Text-Antworten für alle offenen Fragen sind jeweils zusammengefasst.

Auf der letzten Seite befindet sich eine Profillinie im Vergleich zu den Mittelwerten aller Rückläufer für diesen Fragebogen-Typ. Die Profillinie eignet sich auch zur Präsentation in der LV.

Eine Einordnung Ihrer Bewertung ist nach Abschluss der Ergebnisauswertung unter <http://www.tf.fau.de/studium/evaluation> --> Ergebnisse --> SS 13 möglich, hierzu die Bestenlisten, Percentile, etc. einsehen.

Bitte melden Sie an [eva@techfak.uni-erlangen.de](mailto:eva@techfak.uni-erlangen.de) die Anzahl der ausgegebenen TANn, wenn Sie das bis jetzt versäumt haben, Ihnen wurden 292 TANn geschickt.

Mit freundlichen Grüßen

Michael Wensing (Studiendekan, [michael.wensing@ltt.uni-erlangen.de](mailto:michael.wensing@ltt.uni-erlangen.de))  
Jürgen Fricke (Evaluationskoordinator, [eva@techfak.uni-erlangen.de](mailto:eva@techfak.uni-erlangen.de))



## Dipl.-Inf. Jens Schedel

SS 13 • Übungen zu Systemprogrammierung 1  
 ID = 13s-Ü SP1  
 Erfasste Rückläufer = 44 • Formular u\_s13 • LV-Typ "Übung"

### Globalwerte

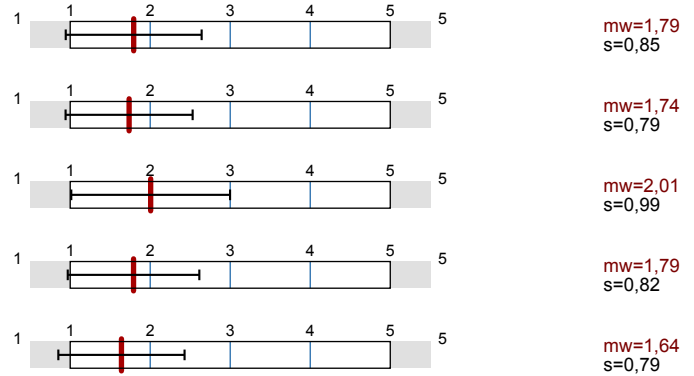
## Globalindikator

Globalfragen für alle Lehrveranstaltungs-Typen (mit Gewichtung)

Übung im Allgemeinen

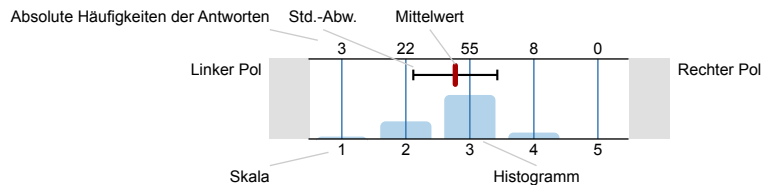
Didaktische Aufbereitung

Präsentation des Übungsleiters



## Legende

Fragetext



n=Anzahl  
 mw=Mittelwert  
 s=Std.-Abw.  
 E.=Enthaltung

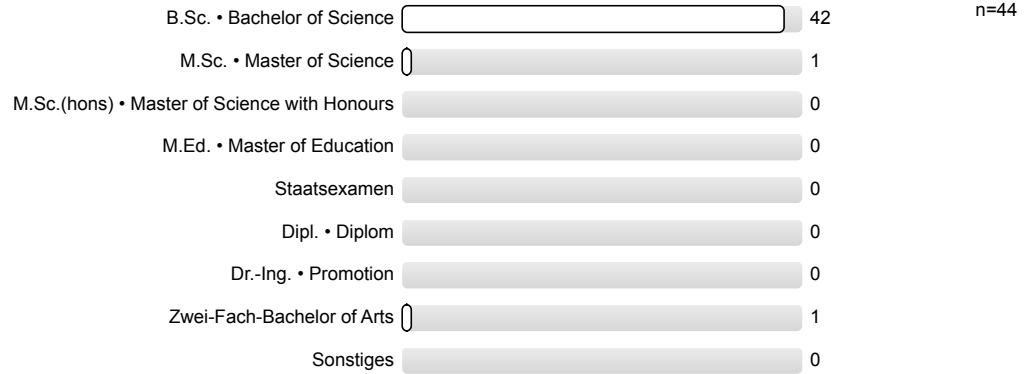
Klick on british flag to get the english survey  
 Achtung: Beim Anklicken der Sprachsymbole verlieren Sie alle bisherigen Eintragungen !

### Allgemeines zur Person

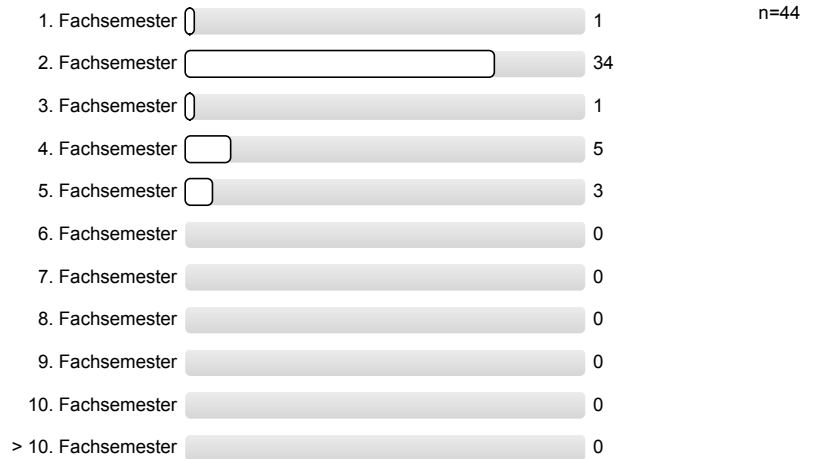
<sup>2\_A)</sup> • Ich studiere folgenden Studiengang:

CE • Computational Engineering	<input type="checkbox"/>	1	n=44
EEl • Elektrotechnik - Elektronik - Informationstechnik	<input type="checkbox"/>	3	
INF • Informatik	<input type="checkbox"/>	29	
IuK • Informations- und Kommunikationstechnik	<input type="checkbox"/>	2	
MT • Medizintechnik	<input type="checkbox"/>	1	
(T)Math • (Techno)-Mathematik	<input type="checkbox"/>	6	
WINF • Wirtschaftsinformatik	<input type="checkbox"/>	2	

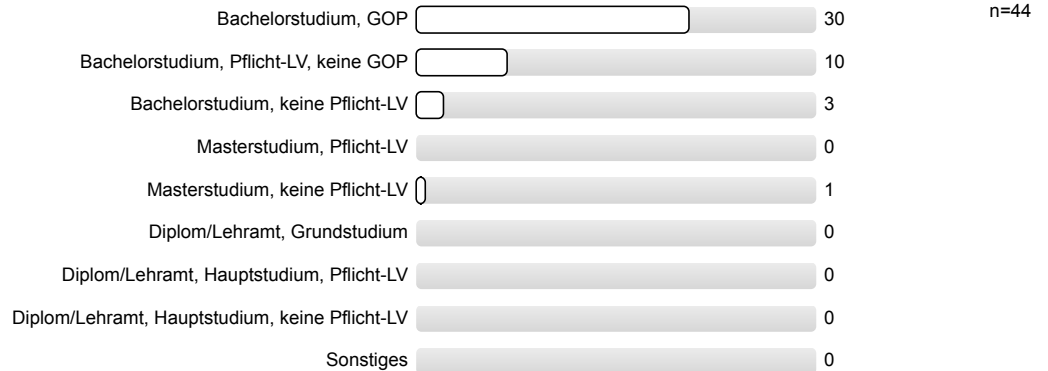
2\_B) • Ich mache folgenden Abschluss:



2\_C) • Ich bin im folgenden Fachsemester:

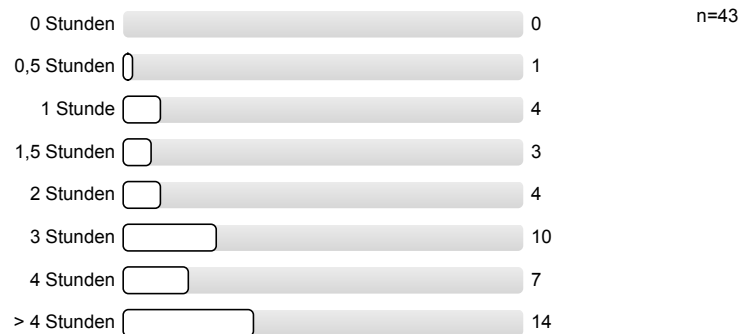


2\_D) • Diese Lehrveranstaltung gehört für mich zum . . . .



Mein eigener Aufwand

3\_A) Mein Durchschnittsaufwand für Vor- und Nachbereitung dieser Übung beträgt pro Übungsstunde (45 Min.):

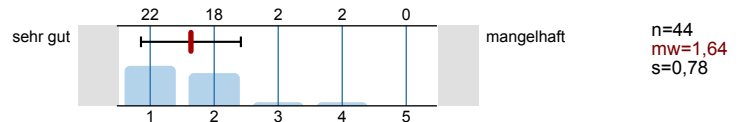


3\_B) Ich besuche etwa . . . Prozent dieser Übung.

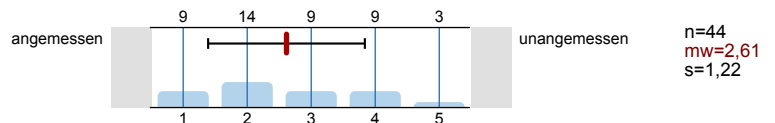


Globalfragen für alle Lehrveranstaltungs-Typen (mit Gewichtung)

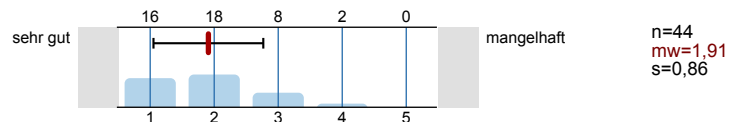
4\_A) • Bitte benoten Sie die Übung insgesamt (50%):



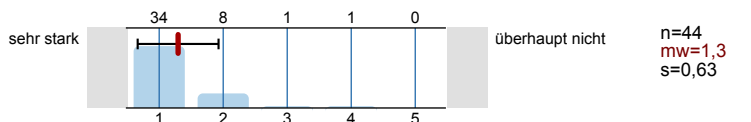
4\_B) • Der notwendige Arbeitsaufwand für diese Übung ist (12,5%):



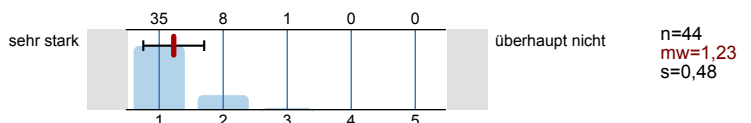
4\_C) • Wie ist die Übung strukturiert (12,5%)?



4\_D) • Der Übungsleiter wirkt engagiert und motiviert bei der Durchführung der Übung (12,5%).

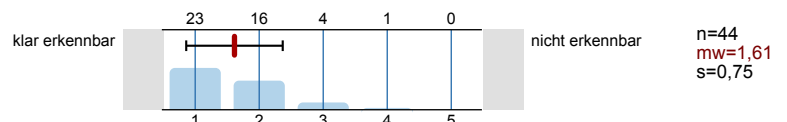


4\_E) • Der Übungsleiter geht auf Fragen und Belange der Studierenden ein (12,5%).

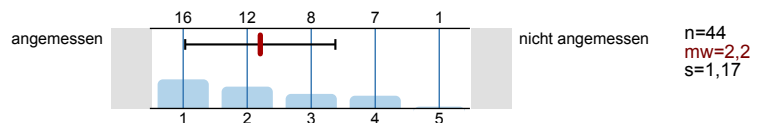


Übung im Allgemeinen

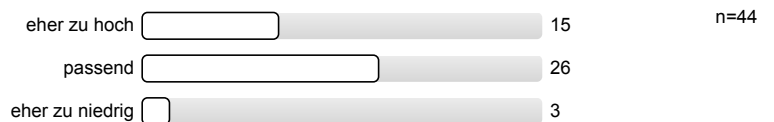
5\_A) Zielsetzungen und Schwerpunkte des Übungsinhalts sind:



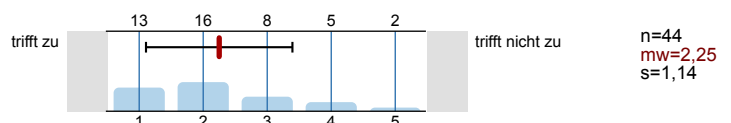
5\_B) Der Schwierigkeitsgrad der Übung ist:



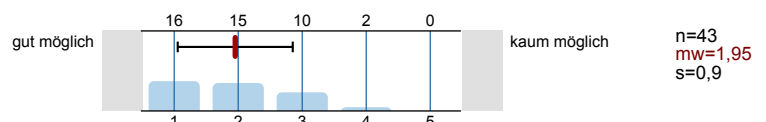
5\_C) Der Schwierigkeitsgrad der Übung ist:



5\_D) Die zur Verfügung gestellten Unterlagen sind in Menge und Qualität den Zielen der Übung angemessen.

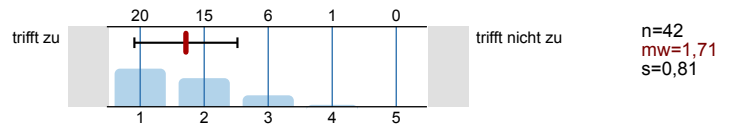


5\_E) Anhand des erarbeiteten Übungsmaterials ist die Vertiefung des Vorlesungs-/Modulinhalts:

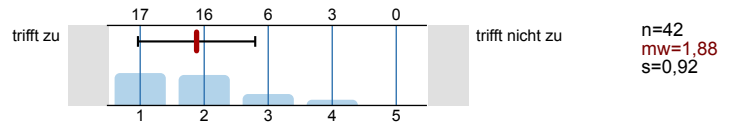


## Didaktische Aufbereitung

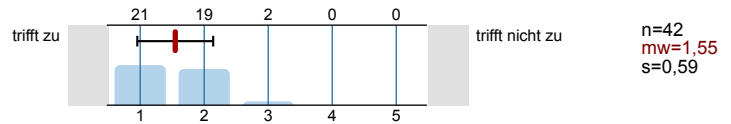
6\_A) Die Präsentation von Aufgaben und Lösungen ist nachvollziehbar, es ist genügend Zeit zum Mitdenken vorhanden.



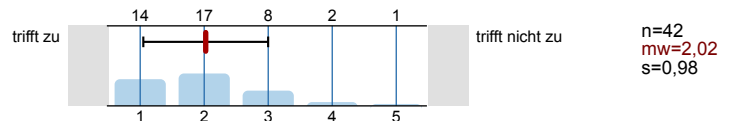
6\_B) Die Anwendbarkeit des Übungsstoffes wird z.B. durch Beispiele gut verdeutlicht.



6\_C) Die Übungsform (Aufgabenbehandlung, Programmieren, etc.) ist gut zur Vermittlung des Stoffes geeignet.

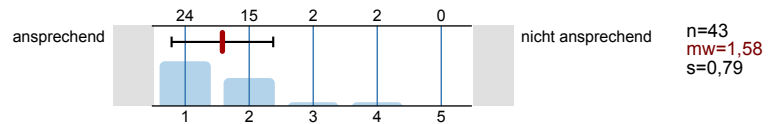


6\_D) Der Bezug zur Vorlesung und den Prüfungsanforderungen wird hergestellt.

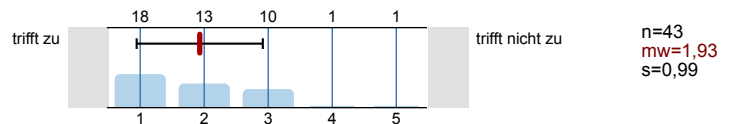


## Präsentation des Übungsleiters

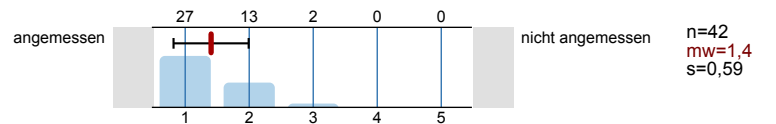
7\_A) Der Präsentationsstil des Übungsleiters ist:



7\_B) Ich werde gut zum selbstständigen Lösen von Aufgaben angeleitet.



7\_C) Der Einsatz und das Zusammenspiel von Medien (Tafel, Overhead-Projektor, Beamer, etc.) ist:



## Weitere Kommentare

9\_A) An der Lehrveranstaltung gefällt mir besonders:

- (T12) netter tutor, Übungsaufgaben werden verständlich erklärt
- Christian Schlumberger ist sehr motiviert uns die aktuellen Themen näher zu bringen. Er ist gut vorbereitet und betreibt einiges an extra Aufwand um den Stoff gut zu vermitteln. Korrekturen sind fair und das gesamte Management der Übung ist so das es Spaß macht. SP ist kein Vergleich zu AUD und das ist auch gut so!
- Christian Schlumberger: Übungsleiter war motiviert und immer gut drauf, sehr gute Übung
- Dass die einzelnen Aufgaben doch recht schön aufeinander aufbauen und teilweise recycled werden. Bei der Halde wäre es z.B. toll gewesen, wenn man seine eigene Lilo - sofern sie funktioniert hat - hätte wiederverwenden können und nur ansonsten auf eure Listenimplementierung zurückgreifen müsste.
- Die Lehrveranstaltung besitzt durch die vielfältigen Programmieraufgaben einen hohen Lernwert! Das sollte so beibehalten werden.
- Die Übungen sind anders als z.B. in AuD nicht Pflicht, aber durch das Bonuspunkte-System ist man trotzdem motiviert die Aufgaben zu lösen, und die Aufgaben machen auch deutlich mehr Spaß. Außerdem ist es gut, dass die Bearbeitungszeit erst mit der Tafelübung beginnt. Wenn man sonst z.B. die SP- und PFP-Übung erst am Ende der Woche hat, wäre es zeitlich nicht machbar, beide Übungen zu bearbeiten.
- Einige Tutoren helfen in der Rechnerübung sehr gut und bringen einen wirklich weiter. Vor allem, dass nahezu immer eine Rechnerübung ist, empfand ich als sehr hilfreich und angenehm
- Gelerntes Anwenden: Die in der Übungsgruppe erarbeiteten Codebeispiele sind sehr hilfreich beim Lösen der Aufgaben. Auch die Referenzimplementierung im pub-Verzeichnis sind nützlich. Da der Abgabetermin relativ zu den Tafelübungen ist, kann man sich die Zeit zum Lösen der Aufgabe gut einplanen.
- Lockere Stimmung in der Übung(Schlumberger)

- Sehr Motiviert  
D.P;
- Sehr locker und ansprechend gehalten.
- Super Übungsleiter ;)
- T16 Max Eschenbacher: Sehr engagiert und motiviert, geht auf Fragen ein, erklärt alles verständlich, macht mit uns zusammen Aufgaben zum besseren Verständnis. Alles tipp topp :D
- T[03] guter Tutor, erklärt alle möglichen Fragen. (selbst wie man im mplayer ein Film mit ASCII-Zeichen abspielt :)
- Tobias Sammet: Super Tutor!
- Vorlesung:  
Bereitstellung der Folien als Ausdruck sowie als pdf-Datei  
graphische Veranschaulichungen  
auftretende Fragen werden berücksichtigt und beantwortet  
eigener Raum zum Programmieren, der fast immer mit Tutoren besetzt ist.  
Man gewöhnt sich an die man-pages  
@Dominik Paulus :  
immer gut drauf und wahnsinnig nett.  
Bringt den Stoff auch mit einem gewissen Elan rüber, den man bei den Vorlesungen manchmal vermisst.  
Erstklassiger Tutor!
- Zur Übung T11: Der Übungsleiter ist sehr engagiert und bringt den Stoff der Übungsaufgaben gut herüber. Er gestaltet seine Tafelübung weder extrem langweilig, noch monoton. Ich hatte bisher selten einen so guten Tutor.
- insgesamt gut abgestimmtes System ( Vorlesung, Übung und Aufgaben )
- Übungsaufgaben immer sehr spannend zu lösen; konsequente Verwendung der deutschen Sprache
- Übungsbetrieb in Systemprogrammierung ist der beste, der mir bisher untergekommen ist. Gut durchdachte und praxisrelevante Aufgaben, viel Freiheit wie man die Aufgaben angeht und löst, und sehr kompetente Tutoren in den Übungen. Oft muss man sich in anderen Übungen durch eine Aufgabenstellung hangeln, die nur darauf hinzielt dass alle das gleiche, der Musterlösung entsprechende, produzieren. In SP kann man sich selbst und frei mit der Materie beschäftigen, was in meinen Augen auch wesentlich produktiver ist und einen wesentlich größeren Lerneffekt hat.  
  
Großes Lob!

9.B) An der Lehrveranstaltung gefällt mir Folgendes weniger, und ich schlage zur Verbesserung vor:

- >svn die!  
Besonders bei den ersten Aufgaben war das Einchecken ins svn wesentlich schwieriger als die Implementierung. Das nervt!  
  
Vergisst man bei der Implementierung drei mal eine Fehlerüberprüfung, gibt's sofort drei Punkte Abzug, obwohl das Programm reibungslos funktioniert. Das ist wirklich ärgerlich. Dass die Hälfte des Codes nur aus Fehlerüberprüfung besteht, macht es nicht übersichtlicher.
- Besonders in der zweiten Hälfte des Semesters kam es zu häufig vor, dass Themen in der Übung besprochen wurden, die noch nicht in der Vorlesung dran waren.  
  
T07: Der "Service" (schnelle Korrektur und ausgedruckte Übungszettel) hat leider nach den ersten Wochen stark nachgelassen.
- Das svn-Abgabesystem nervt das eine oder andere Mal, wenn es zum x-ten Mal irgendeine Fehlermeldung ausspuckt, weil wieder irgend etwas nicht passt. Und wenn man die Aufgaben sowieso mit seinem Gruppenpartner vor einem Rechner macht, ist auch der Großteil des Sinnes dahinter dahin. Dann wäre das EST des Lehrstuhls 2 deutlich komfortabler  
Außerdem ist es EXTREM ÄRGERLICH, wenn man es geschafft hat - trotz Windhund-Prinzip (was so ziemlich eines der unfairsten Zuteilungsverfahren ist, abgesehen von einer rein zufälligen Zuteilung) - sich in seine Wunschübung einzutragen, in der nächsten Vorlesung zu erfahren, dass da was schiefgelaufen ist und man sich bitte nochmal anmelden soll, aber nur diejenigen, bei denen es nicht geklappt hat. Dass dann natürlich schon fast alle Termine belegt sind, stört aber keinen und man soll sich halt einfach irgendwo dazuschreiben, wo noch Plätze frei sind. Das wäre ja alles halb so wild, wenn man nicht GENAU DIESE Tafelübung besuchen müsste und die Gruppenaufgaben auch nur mit einem Partner aus eben dieser Übung machen dürfte. Deswegen auch an dieser Stelle nochmal der Rat: STEIGT AUFS EST DES LEHRSTUHL 2 UM, damit soetwas nicht noch einmal passiert.
- Die Besprechungs-Übungen fand ich manchmal ein bisschen kurz, vlt. könnte man da noch ein paar allgemeine Dinge aus der Vorlesung wiederholen (z.B. anhand von Fragen die man auch in der Prüfung erwarten könnte, besprechen), da man in den anderen Übungen ja primär auf die Hausaufgabe vorbereitet wird.
- Die Programmierung nimmt oft sehr viel mehr Zeit in Anspruch! Die Credit-Points und der damit veranschlagte Workload wird deutlich überschritten, somit sollte eine Anhebung der Credit-Points erfolgen.
- Die Übungsaufgabe zu Beginn etwas leichter zu gestalten, um den Einstieg in C zu erleichtern
- Evtl sollte darüber nachgedacht werden, pro Rechnerübung 2 Tutoren einzuteilen, da man teilweise schon bis zu einer halben Stunde warten musste bis man Hilfe bekam und wenn es ein essentielles Problem war, dass ein Weiterarbeiten verhindert, ging da schon viel Zeit verloren.  
Beschwerden möchte ich mich über die Rechnerübung Di von 16:00 bis 18:00 Uhr, da ich die komplette Zeit über anwesend war, und sich in der gesamten Zeit niemand als Tutor zu erkennen gab und als ich dann eine Frage hatte auch niemand mehr im Cip war, der helfen

konnte...

- Ich finde den Zusammenhang zwischen Vorlesung und Übung nicht angemessen. Mit kommt es so vor, als ob in der Vorlesung einfach nur Bereiche angekratzt werden und dann nebenbei alles mal im Detail implementiert werden muss.
- Ich habe jetzt nach dem ersten Semester irgendwie noch nicht wirklich viel über Systemprogrammierung gelernt. Ich war zwar nie in der Vorlesung, aber vielleicht könnte man auch thematischen Stoff mit Wissensfragen abfragen/in der Übung behandeln
- Ich hatte da jetzt zwar keine grossartigen Probleme mit, aber es ist mir bei einigen anderen aufgefallen, dass die doch grosse Probleme mit dem Einhalten der Schnittstellerspezifikation und der vollstaendigen Fehlerbehandlung hatten.  
Es ware toll gewesen, wenn in einer der ersten Uebungen das Manpages Lesen, Umsetzen und die Fallstricke, die es dabei gibt (man liest ja doch oeffter mal "es sollte sich so verhalten") staerker Thematisiert worden waren. Das hatte ich am Anfang dann doch ein bisschen unterschaezt, die Manpages erstmal nur ueberflogen, und dann gleich Punktabzug fuer einen ueberlesenen Sonderfall bekommen. Klar, das macht man nur ein, zwei mal, weils verdammt aergerlich ist.. aber ihr wisst schon was ich meine.. hoff ich :)
- Ohne gute C/Linux Vorkenntnisse sprengen die Hausaufgaben gerne mal den zeitlichen Rahmen.  
Ein Vorkurs vor den/während der ersten Vorlesungswochen (wie in Java für AuD) würde das Fach deutlich leichter gestalten und es auch Anfängern ermöglichen Punkte in den Übungen zu sammeln.
- Rauer Einstieg in Sachen Konsolennutzung, Subversion. Ein kleines Trainingsszenario in der 1. Übung wäre vielleicht effektiver als ein Foliensatz mit einer Kurzfassung über alle Werkzeuge für die Übung + sämtliche Begriffe, die einen am Anfang glatt erschlagen.
- Sowohl der Vorlesung als auch der Übung mangelt es stark an Codebeispielen. Oft ist es sehr mühsam aus den Übungsfolien brauchbare Informationen herauszuholen. Herauszufinden, wie man ein Makefile schreibt, war z.B. reine Google-Arbeit. Da hätte man ruhig ein komplettes Beispiel geben können. Den Aufgabenstellungen könnte man noch Implementierungsideen hinzufügen, wie man an den Abgabebildern sieht, sind sie schon recht schwer. Bei der Korrektur wird teilweise übertrieben pingelig vorgegangen: Es werden Dinge bemängelt, die teilweise nicht einmal so von Bibliotheksbefehlen erfüllt werden.
- Spezifikationen sind ungenau, widersprüchlich, nicht vollständig oder nicht als solche gekennzeichnet. Bitte exakt angeben, welche Dokumente und Textstellen Spezifikation und welche Hilfestellung sind.  
Vorbild sollten RFCs und W3C sein.
- Tatsächlicher Vorlesungsinhalt wird zu wenig vertieft. Man könnte beispielsweise zusätzlich zur Programmieraufgabe ein oder zwei niedrig bewertete Fragen zum Vorlesungsstoff stellen, deren Lösung als PDF ebenfalls über svn abgegeben werden kann.
- Vielleicht zu den Aufgaben ein paar Hinweise, dass bestimmte Sachen so nicht optimal sind und man es anders schöner machen könnte. Oder unbewertete Sachen die man noch freiwillig hinzufügen könnte, wenn es einen interessiert. Mit solchen Sachen könnte man den Denkprozess noch ein bisschen anregen und den Blick für bestimmte Punkte vielleicht von Anfang an schärfen.
- bezug der übungen zum vorlesungsstoff teils schwer erkennbar  
im der bewertugn der übungsaufgaben teils strenge bewertungen, viel pnktabzutg auf kleinere Fehler
- quasi-Anwesenheitspflicht bei der Hausaufgaben Besprechung. Wenn man einmal nicht zur eigenen Übung kann, sollte man nicht 0 Punkte bekommen.
- svn Skripte waren unzuverlässig und launisch. Eine bessere Dokumentation zur Verwendung wäre schön
- Übung J.Schedel: In der Übung wird kaum auf evtl. Probleme bei der Aufgabenstellung eingegangen, was das Lösen der Aufgaben unnötig erschwert, bzw. werden dann in der Korrektur (durch anderen Ü.leiter) Dinge bemängelt, auf die in der Übung nicht eingegangen wurde. Im übrigen wären mehrere kleine Aufgaben besser bearbeitbar als eine große Aufgabe.  
Schade :-)

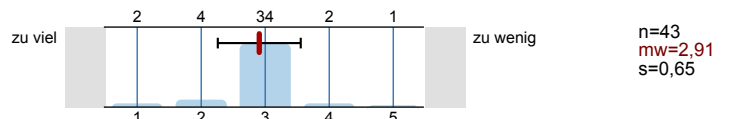
9.C) Zur Lehrveranstaltung möchte ich im Übrigen anmerken:

- Christopher Bross
- Das schöne an den Aufgaben ist, dass sie sehr anwendungsnah sind, aber wenn man grade erst C als neue Sprache lernt, sind die Aufgaben doch extrem anspruchsvoll und zeitaufwendig.
- Die Folien sind teilweise etwas Text-lastig, lässt sich aber bei einigen Themen für das Verständnis nicht vermeiden.  
Wer Vorkenntnisse besitzt hat in dieser Vorlesung definitiv einen Vorteil. Als reiner C-Neuling ist der Einstieg relativ schwer, nicht nur in die Sprache selber, bei der man nach ein paar Übungen den Überblick über alle vorgestellten Funktionen verliert, sondern auch durch die Nutzung von Subversion und der Konsole. Jedoch profitiert man von den neuen Kenntnissen und lernt die man-pages wertzuschätzen.
- Die Umstellung des Semesterplans von dieser einen, unuebersichtlichen Liste zu einem Plan fuer jede TUE war sehr angenehm :)  
Danach hat man sich ohne zu ueberlegen problemlos zurechtgefunden.
- Halde ist echt gross geworden. Zu viel neues auf einmal (gdb, makefile, nicht einfache funktionen wie malloc). Finde ich fuer als dritte aufgabe zu umfangreich. Crawl ist z.b. dafuer im vergleich zu klein.
- Mehr kleinere Übungsaufgaben als 6 große Aufgaben in diesem Umfang
- Mercurial ist viel schöner als SVN ;)
- Sinnvollste Veranstaltung im 2. Semester CE!

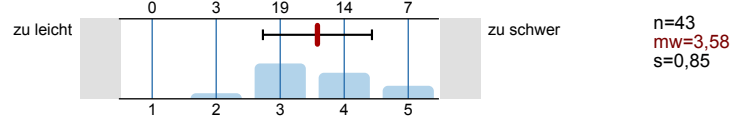
- Vorlesungen immer interessant gestaltet; Folien übersichtlich; Professoren wirken motiviert;

Optionale Zusatzfragen des Übungsleiters

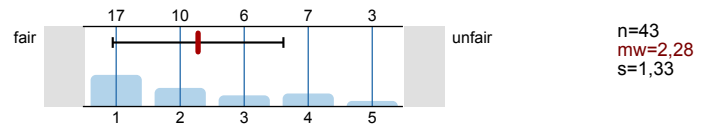
<sup>10\_A)</sup> Anzahl der Übungsaufgaben



<sup>10\_B)</sup> Schwierigkeit der Übungsaufgaben

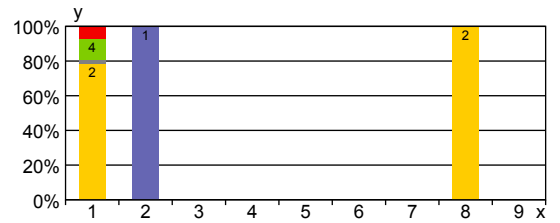


<sup>10\_C)</sup> Korrektur





	y											
x	1	2	3	4	5	6	7	8	9	10	11	
1	0	33	1	5	3	0	0	0	0	0	0	42
2	1	0	0	0	0	0	0	0	0	0	0	1
3	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0
8	0	1	0	0	0	0	0	0	0	0	0	1
9	0	0	0	0	0	0	0	0	0	0	0	0
	1	34	1	5	3	0	0	0	0	0	0	44



x: • Ich mache folgenden Abschluss:

- 1: B.Sc. • Bachelor of Science
- 2: M.Sc. • Master of Science
- 3: M.Sc.(hons) • Master of Science with Honours
- 4: M.Ed. • Master of Education
- 5: Staatsexamen
- 6: Dipl. • Diplom
- 7: Dr.-Ing. • Promotion
- 8: Zwei-Fach-Bachelor of Arts
- 9: Sonstiges

y: • Ich bin im folgenden Fachsemester:

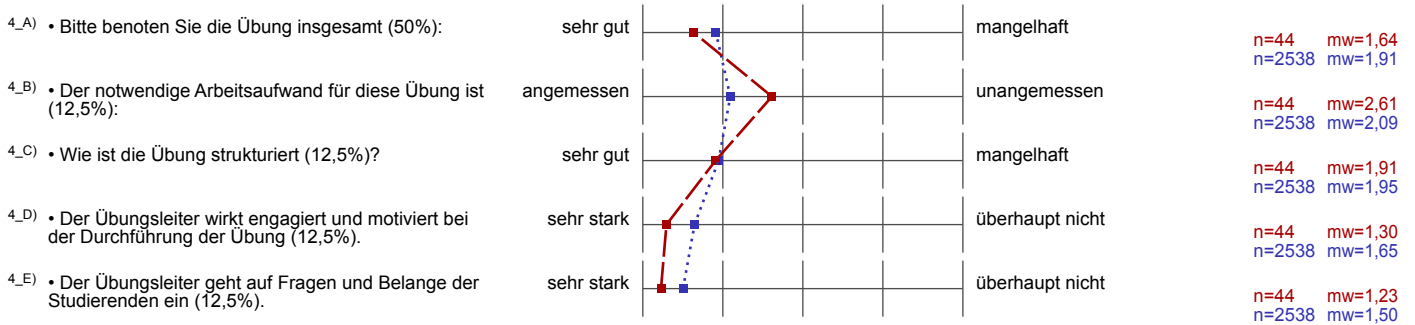
- 1: 1. Fachsemester
- 2: 2. Fachsemester
- 3: 3. Fachsemester
- 4: 4. Fachsemester
- 5: 5. Fachsemester
- 6: 6. Fachsemester
- 7: 7. Fachsemester
- 8: 8. Fachsemester
- 9: 9. Fachsemester
- 10: 10. Fachsemester
- 11: > 10. Fachsemester

# Profillinie

Teilbereich: Technische Fakultät (TF)  
 Name der/des Lehrenden: Dipl.-Inf. Jens Schedel  
 Titel der Lehrveranstaltung: Übungen zu Systemprogrammierung 1 (13s-Ü SP1)  
 (Name der Umfrage)  
 Vergleichslinie: Mittelwert aller Übungs-Fragebögen im SS 13

Verwendete Werte in der Profillinie: Mittelwert

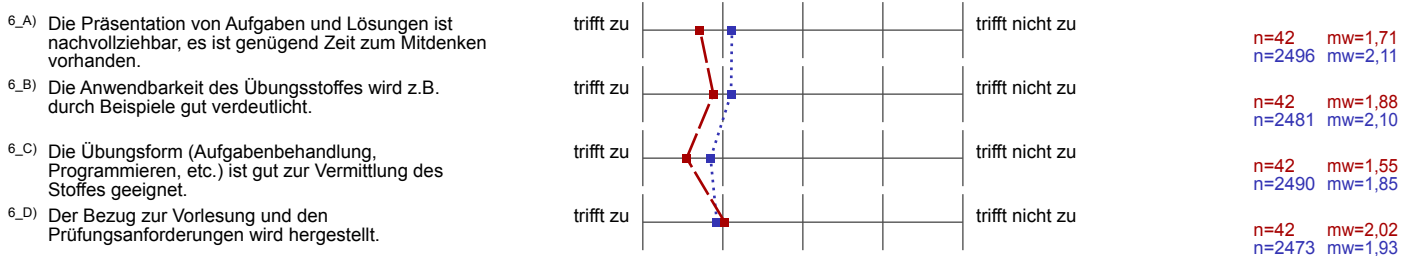
## Globalfragen für alle Lehrveranstaltungs-Typen (mit Gewichtung)



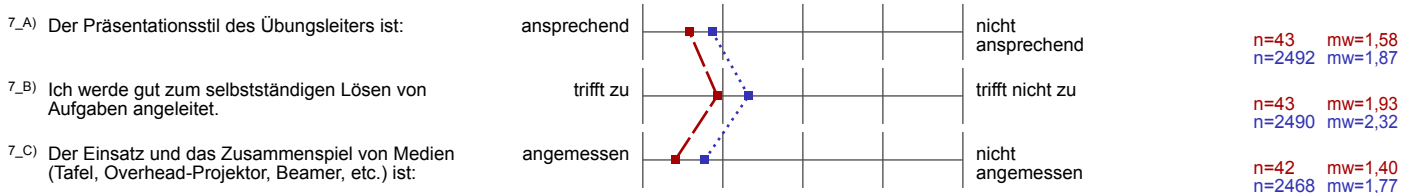
## Übung im Allgemeinen



## Didaktische Aufbereitung

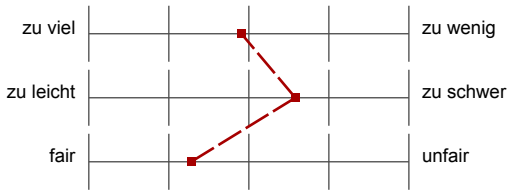


## Präsentation des Übungsleiters



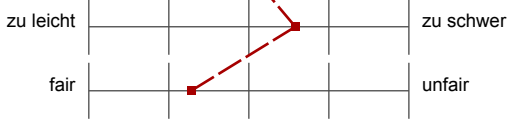
Optionale Zusatzfragen des Übungsleiters

10\_A) Anzahl der Übungsaufgaben



n=43 mw=2,91

10\_B) Schwierigkeit der Übungsaufgaben



n=43 mw=3,58

10\_C) Korrektur



n=43 mw=2,28