



FAU • Dekanat der TF  
Martensstraße 5a  
91058 Erlangen

FAU • Dekanat der TF • Martensstr. 5a • 91058 Erlangen

Herr  
Dr.-Ing. Jürgen Kleinöder  
(PERSÖNLICH)

## SS23 • Systemprogrammierung 1 - Übung

Sehr geehrter Herr Dr.-Ing. Kleinöder,

im Rahmen der Lehrveranstaltungsevaluation im SS23 erhalten Sie hiermit die Auswertung zu Ihrer Umfrage vom Typ "Übung":

- Systemprogrammierung 1 - Übung -

Es wurden hierfür 39 Fragebögen vom Typ "t\_s23\_v+ü1" von den Studierenden ausgefüllt.

Die 4 Indikatoren zeigen den mit der Anzahl der Antworten gewichteten Mittelwert der Skalafragen in den genannten Fragenkapiteln.

Der Mittelwert der 4 Indikatoren bildet den Globalindikator bzw. den Lehrqualitätsindex (LQI).

Für die Einzelfragen und Indikatoren kennzeichnet der Wert 1 hierbei eine maximale Güte, der Wert 5 eine minimale Güte.

Bei den Einzelfragen werden je nach Fragen-Typ die Anzahl und Verteilung der Antworten, Mittelwert und Standardabweichung aufgelistet.

Die Text-Antworten für jede offene Frage sind zusammengefasst aufgelistet.

Die Profillinien zeigen den Vergleich zu den Mittelwerten aller Rückläufer der Technischen Fakultät.

Der LQI und die Indikatoren werden bei genügend (ab 5) Rückläufern zur Qualitätssicherung durch die Studienkommissionen und die Erstellung der Bestenlisten verwendet.

Mit freundlichen Grüßen

Rolf Wanka (Studiendekan, [tf-studiendekan-lehre@fau.de](mailto:tf-studiendekan-lehre@fau.de))  
Jürgen Frickel (Evaluationskoordinator, [tf-evaluation@fau.de](mailto:tf-evaluation@fau.de))

Dr.-Ing. Jürgen Kleinöder

Systemprogrammierung 1 - Übung

Umfragen-Periode: SS23 • LV-Typ: Übung • Rückläufer: 39

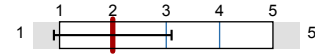
LV-ID: 23s-344141 • Fragebogen-ID: t\_s23\_v+ü1



Globalwerte

## Globalindikator

Indikator • Organisation, Inhalte und Kompetenzen der LV (Kap. 3)



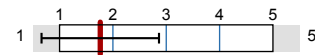
mw=2  
s=1,1

Indikator • Struktur der LV (Kap. 4)



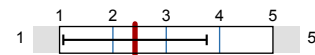
mw=1,83  
s=0,95

Indikator • Durchführung der LV (Kap. 5)



mw=1,98  
s=1,02

Indikator • Zufriedenheit und Kompetenzerwerb (Kap. 6)

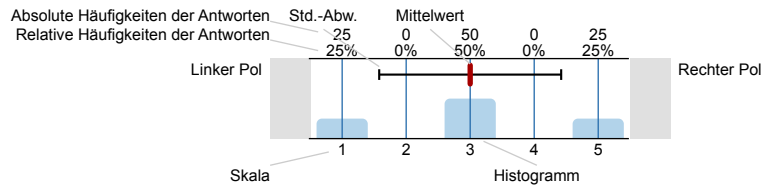


mw=1,76  
s=1,1

mw=2,41  
s=1,34

## Legende

Fragetext



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

## 2. Studierender und Lehrveranstaltung

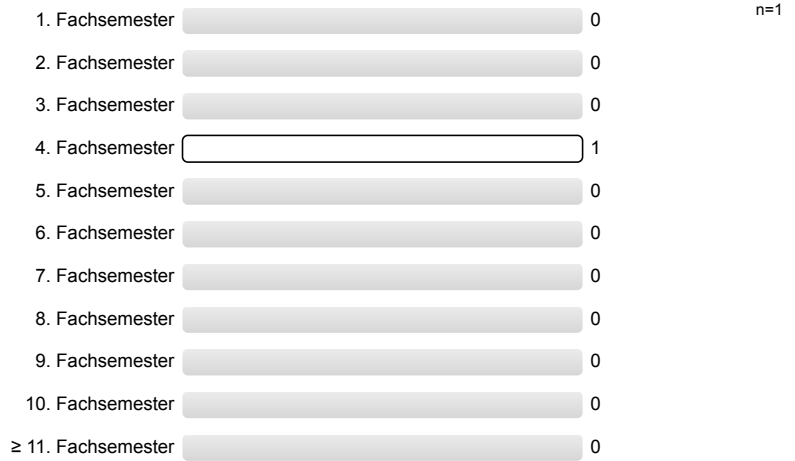
2.1) Ich studiere folgenden Studiengang und Abschluss:

CE • Computational Engineering (B.Sc.)	<input type="checkbox"/>	2	n=39
INF • Informatik (B.Sc.)	<input type="checkbox"/>	30	
INFLA • Informatik für Lehramt (Staatsexamen)	<input type="checkbox"/>	1	
IuK • Informations- und Kommunikationstechnik (B.Sc.)	<input type="checkbox"/>	1	
MT • Medizintechnik (B.Sc.)	<input type="checkbox"/>	3	
TM • Technomathematik (B.Sc.)	<input type="checkbox"/>	1	
Sonstiges	<input type="checkbox"/>	1	

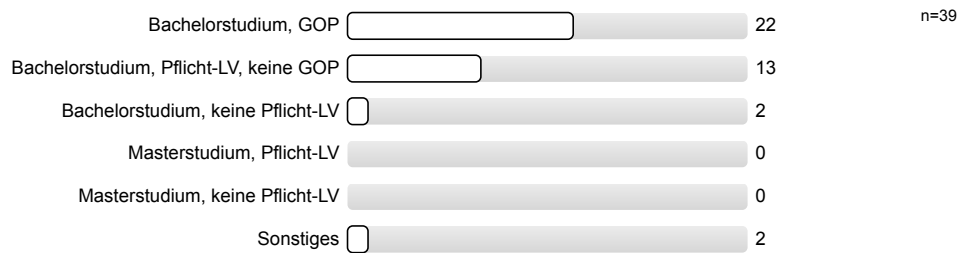
2.2) Ich bin im folgenden Fachsemester (im Bachelor):

1. Fachsemester	<input type="checkbox"/>	0	n=37
2. Fachsemester	<input type="checkbox"/>	27	
3. Fachsemester	<input type="checkbox"/>	1	
4. Fachsemester	<input type="checkbox"/>	6	
5. Fachsemester	<input type="checkbox"/>	0	
6. Fachsemester	<input type="checkbox"/>	3	
7. Fachsemester	<input type="checkbox"/>	0	
8. Fachsemester	<input type="checkbox"/>	0	
9. Fachsemester	<input type="checkbox"/>	0	
≥ 10. Fachsemester	<input type="checkbox"/>	0	

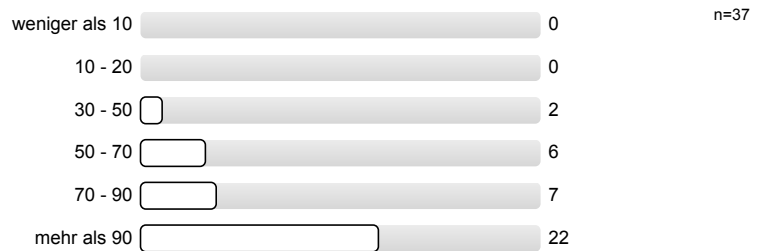
2.4) Ich bin im folgenden Fachsemester (im Staatsexamen):



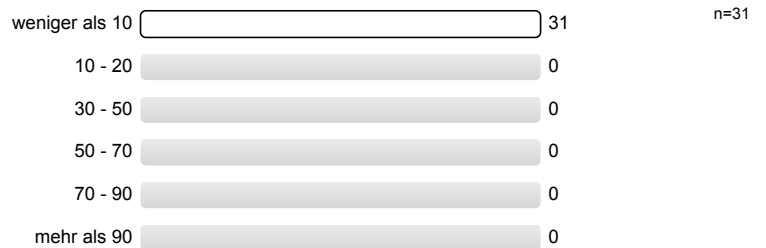
2.5) Diese Lehrveranstaltung (LV) gehört für mich zum . . . .  
 ("keine Pflicht-LV": Wahl einer anderen, alternativen LV wäre möglich)



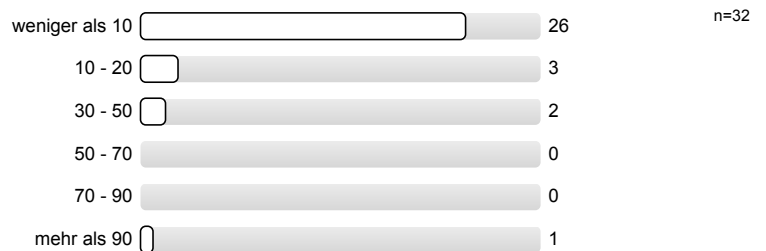
2.7) An . . . . Prozent dieser LV habe ich synchron in Präsenz teilgenommen.



2.8) An . . . . Prozent dieser LV habe ich synchron online (Zoom, Teams, o. ä.) teilgenommen.

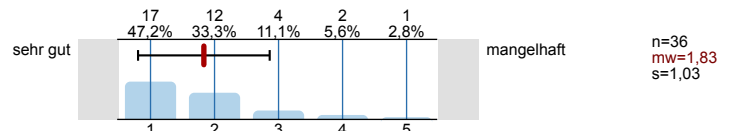


2.9) An . . . . Prozent dieser LV habe ich asynchron (Aufzeichnung, o. ä.) teilgenommen.

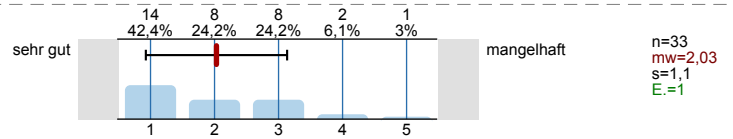


### 3. Organisation, Inhalte und Kompetenzen der Lehrveranstaltung

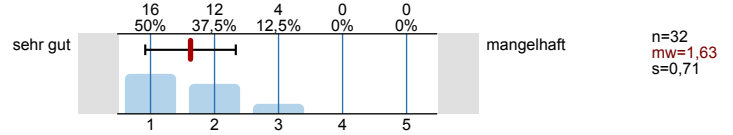
3.1) Wie gut war die Durchführung der LV organisiert?



3.2) Wie gut war die LV inhaltlich organisiert und mit evtl. zugehörigen LVen abgestimmt (Vorl. • Übg. • Prakt. • ...)?

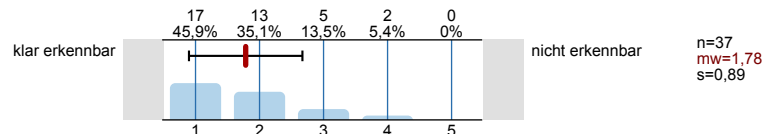


3.3) Die LV entspricht den im Modulhandbuch eingetragenen Inhalten und Kompetenzen.

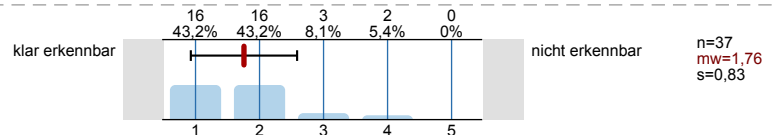


### 4. Struktur der Lehrveranstaltung

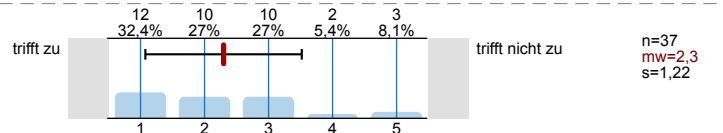
4.1) Zielsetzungen und Schwerpunkte des Inhalts waren:



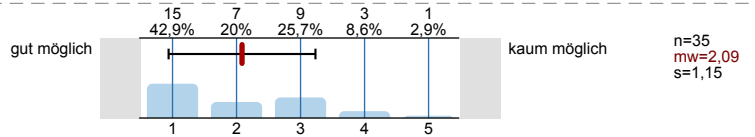
4.2) Der rote Faden der LV (synchron bzw. asynchron) war:



4.3) Der dargebotene Stoff war nachvollziehbar, es war genügend Zeit zum Mitdenken vorhanden.

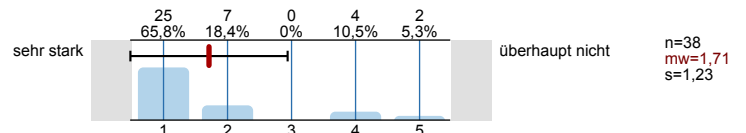


4.4) Mit den Medien, Begleitmaterialien, Literaturhinweisen und Hinweisen in der LV selbst waren Vor- und Nachbereitung:

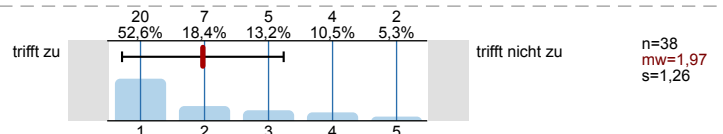


### 5. Durchführung der Lehrveranstaltung

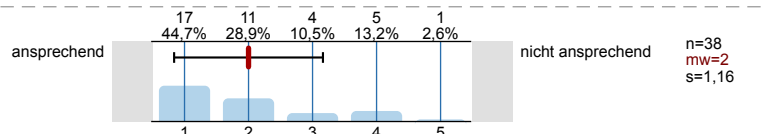
5.1) Die Dozentin/Der Dozent wirkte engagiert und motiviert bei der Durchführung.



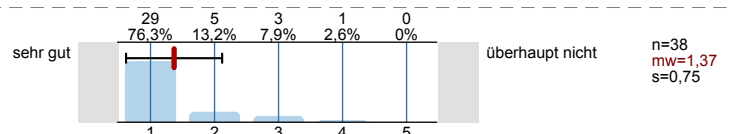
5.2) Die Dozentin/Der Dozent förderte das Interesse am Themenbereich.



5.3) Der Präsentationsstil der Dozentin/des Dozenten war:

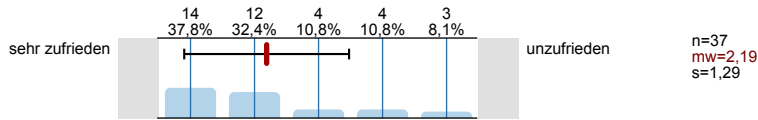


5.4) Die Dozentin/Der Dozent ging auf Fragen und Belange der Studierenden ein (synchron und asynchron).

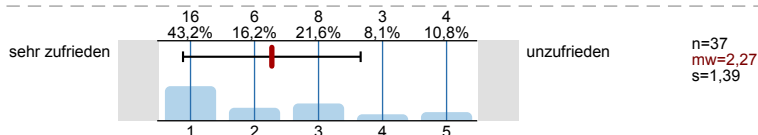


### 6. Zufriedenheit und Kompetenzerwerb

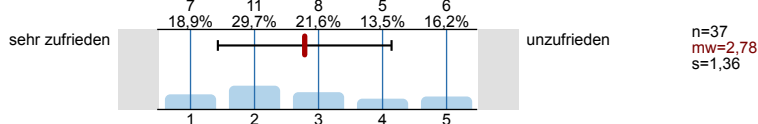
6.1) Wie zufrieden sind Sie insgesamt mit der LV?



6.2) Wie zufrieden sind Sie mit der LV bezüglich Ihres eigenen Kompetenzerwerbs?

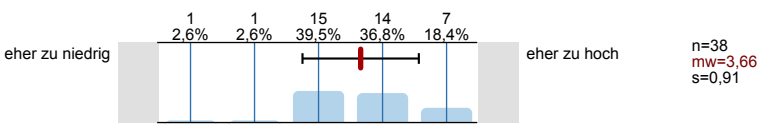


6.3) Wie zufrieden sind Sie mit dem Verhältnis zwischen Lernerfolg/Kompetenzerwerb und eigenem Zeitaufwand?

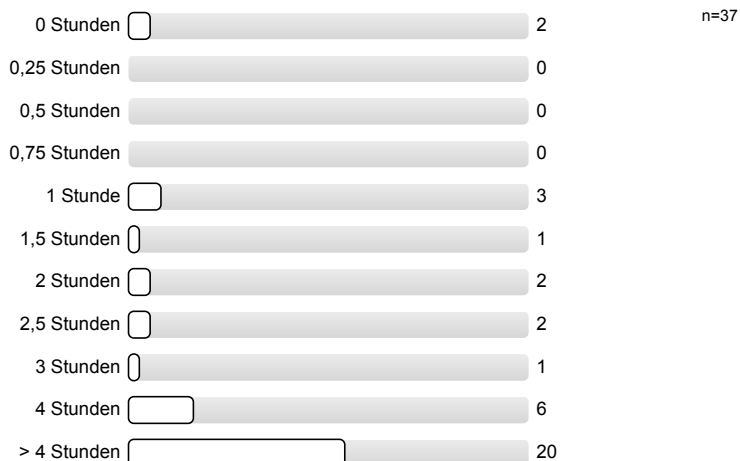


7. Zusätzliche Informationen für die Dozentin/den Dozenten

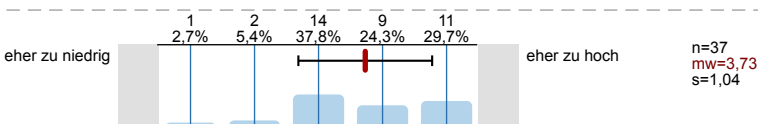
7.1) Der Schwierigkeitsgrad des Stoffes war für mich:



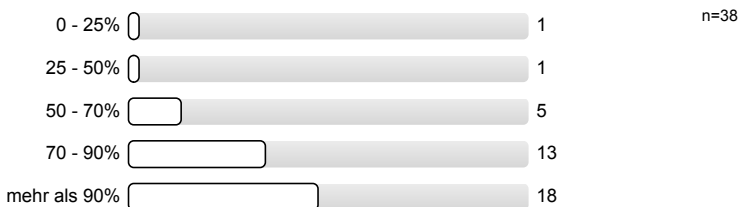
7.2) Mein Durchschnittsaufwand für Vor- und Nachbereitung dieser LV (ohne den LV-Besuch) betrug pro Woche:



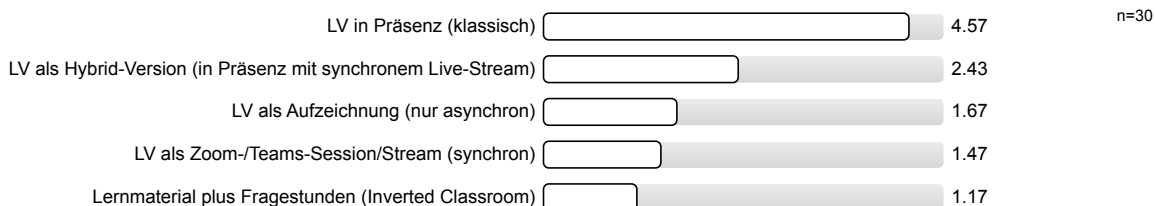
7.3) Meinen zeitlichen Durchschnittsaufwand für diese LV fand ich:



7.4) Ich habe bei etwa . . . . Prozent dieser LV zeitnah (mit höchstens 1 Woche Verzug) mitgearbeitet.



7.5) Welche Darbietungs-Form hätten Sie sich für diese LV gewünscht (bitte mit Drag-and Drop nach Priorität sortieren)?



7.6) An der Durchführung der Lehrveranstaltung gefielen mir folgende Aspekte besonders:

- ----
- CI war eigentlich ganz gut, vor allem bei der halde, die ja eine reine libraryaufgabe (zwar mit tests, aber trotzdem), und damit grobe "Leichtsinnfehler" verhindert wurden
- Der Tutor war besonders erfahren
- Die alten Aufzeichnungen waren teilweise sehr hilfreich beim bearbeiten der Aufgaben. Das kurze besprechen der Aufgabe in den jeweiligen Uebungen (also vor der Abgabefrist) um einen kurzen ueberblick ueber die Aufgabe zu erhalten war sehr gut.
- Die Aufgaben waren sehr vielfältig und gut zu bearbeiten. Die Abgabe über GitLab konnte direkt zum Üben bzw. Erlernen einer Versionsverwaltung genutzt werden.
- Die Einbindung von Git ist wirklich gut. Nur durch aktive Verwendung lernt man so ein System.
- Die neu eingeführten Gitlab Tests
- Felix richtig korrekt, sehr guter Tutor
- Gut durchdachte und gestellte Aufgaben. Das neue Git System funktioniert super.
- Gute Anwendungsbezogene Beispiele, klare Differenzierung zwischen Verwendungsweisen in der Praxis und Theorie. Sehr gute weiterführende Informationen.
- Gute Organisation via Git; Engagierte Tutoren
- Herr Preibsch bemüht sich nach Kräften, die Studenten für die inhärent demotivierenden Hausaufgaben zu begeistern. Er erklärt verständlich, sorgt für eine lockere Atmosphäre und beweist Erfahrung als Tutor. Auch bei Fragen zur Korrektur hilft er weiter.
- Interaktive Gestaltung der Übungen, Erreichbarkeit und Hilfsbereitschaft außerhalb der Übungen, zusätzlich zur Verfügung gestelltes Material, Erklärungen verschiedener Herangehensweisen an die Aufgaben, sehr umfangreiches Fachwissen des Dozenten, ausführliche und hilfreiche Bewertungen zu den Programmieraufgaben
- Lockerer, sympatischer Typ.
- Mir gefiel, dass es beim C-Teil noch Aufzeichnungen gab.
- Organisation mit Besprechung und Vorstellung der Themen im Wechsel.
- Toller Tutor!
- Umfassendes Wissen des Übungsleiters; interessante Zusatzinfos
- Wenn der Stoff nachvollziehbar war sehr interessant.
- Übungen waren interessant, insbesondere Anekdoten und Zusatzinformationen

7.7) An der Durchführung der Lehrveranstaltung gefiel mir Folgendes weniger, und ich schlage zur Verbesserung vor:

- - kein Übungsbetrieb im eigentlichen Sinn, sondern nur Vorstellen von Aufgaben (denen, die es gemacht haben bringt es keinen Mehrwert und denen, die es nicht gemacht haben hilft es nicht bei der Entwicklung eigener Lösungen) und Vermittlung neuen Stoffs durch den Tutor (wieso ist der Tutor dafür zuständig und nicht die Vorlesung??)  
-> kleine "Miniaufgaben", die in Präsenz bearbeitet + besprochen werden sinnvoll, auch als Vorbereitung für die Hausaufgabe
- - Tutor konnte oft fachliche Fragen nicht beantworten  
- Tutor konnte Korrektur teils nicht begründen  
- allgemein zu strenge Korrektur (z.B. Folgefehler quasi nicht existent) (dies wurde auch in vergangenen Evaluationen angekreidet, aber nicht verbessert!)
- - zu hoher Zeitaufwand für die Aufgaben, sodass man bei anderen Modulen in Verzug geraten würde  
- nicht anfängerfreundlich und in diesem Format nicht fürs 2. Semester geeignet, hoher Punktabzug entmutigt zusätzlich  
- Vorschlag: Tafelübungen sollten nicht im Vordergrund stehen. Stattdessen bitte mehr Termine für betreute Rechnerübungen anbieten, da diese tatsächlich das Verständnis fördern und nicht nur irgendwelche Funktionsnamen an die Stirn werfen. Da man sowieso schon externe Webseiten wie Stackoverflow oder die manpages zu Rate ziehen soll, braucht man keine halbherzige, für manche Übungsaufgaben zu oberflächliche Erklärungen.
- An Tagen, an denen nur eine Aufgabe zu besprechen ist, endet die Übung immer nach dieser. Ich komme also für gerade mal 20 Minuten in die Übung und kann danach wieder nach Hause fahren... Wäre es nicht möglich trotz Vorstellung noch weiteren Stoff zu besprechen und evtl andere Übungen so zu entzerren?
- Aufgabenstellungen mangelten teils an Detail, z.B. bei Mach die Synchronisierung war nicht immer klar.
- Bitte nicht nur beim C-Teil Aufzeichnungen, sondern auch bei den anderen Teilen!
- Da ich keine Vorkenntnisse in C hatte, war mein eingebrachter Aufwand ziemlich hoch, verglichen mit anderen Veranstaltungen
- Das Tempo war hoch für 2.-Semestler. Die Abgabefristen würden mit Abgabefristen anderer Veranstaltungen teilweise überlappen,

was aber verständlich für eine Veranstaltung für 2 verschiedene Fachsemester ist.

- Der Tutor (erstes mal SP Tutor) konnte leider machmal nicht auf die eigenen Korrekturhinweise eingehen. Generell ist die Zielsetzung der Übungsaufgaben völlig am Ziel vorbei, es wird viel zu wenig Wert auf das Verständnis von C gelegt, stattdessen bekommt man für alles Abzüge, die in dem Maße wohl kein Student versteht. Traurig, dass sich der Lehrstuhl die Kritik, die zumindest laut den Tutoren schon seit Jahren existiert, sich nicht zu Herzen nimmt. Dann kann man eine Evaluation auch sein lassen...
- Die Hausaufgaben sind wirklich extrem ehrenlos. Entweder man macht einen verpflichtenden Übungsschein, dann aber mit moderatem Anspruch und Aufwand, da es alle machen müssen, oder man macht, wie aktuell, freiwillige Hausaufgaben für die Interessierten, dann aber so, dass die Aufgaben motivieren. Die aktuellen Aufgaben sind für den Bachelor-Pflichtbereich viel zu schwer und zeitaufwändig und die Korrekturrichtlinien, für die die Tutoren nichts können, viel zu strikt. Solchen Aufgaben kann man in anspruchsvollen Vertiefungsveranstaltungen stellen - für Bachelor-Studenten des 2. Semesters sind sie reiner Hohn und bestrafen diejenigen, die sich für das Fach interessieren.
- Die Kluft zwischen den C-Könnern\*innen wird durch die übermäßig schweren Aufgaben weiter verstärkt.

Warum setzt ihr nicht auf ein paar zusätzliche, etwas leichtere Übungsaufgaben anstelle von weniger schweren? Vielleicht könntet ihr wöchentliche Abgaben einführen, bei denen alle mitgenommen werden und niemand auf der Strecke bleibt. Ihr könntet beispielsweise die komplexen Aufgaben in mehrere Teile aufteilen, sodass selbst wenn man nicht alles schafft, immer noch Punkte vergeben werden. Die derzeitigen Aufgaben sind einfach unnötig schwierig, und die drastischen Punktabzüge für Fehler, die unvermeidbar sind, wenn man zum ersten Mal C programmiert, wirken sehr demotivierend. Fehler dienen dem Lernen, aber hier in SP scheinen Fehler eher dazu da zu sein, um uns zu bestrafen, wenn wir Fehlerüberprüfungen vergessen. Ich finde das pädagogisch fragwürdig und es stellt den Bildungsauftrag, den ihr habt, in Frage, den Studierenden etwas beizubringen.

Dieser C-Crashkurs führt vor allem zu einem regelrechten Crash und einer Menge Frustration in den Übungen. Wenn sowieso alles in den Übungen um die Programmierung in C dreht, warum bietet ihr dann nicht einfach einen fundierten C-Programmierungskurs mit entsprechenden Übungen an, um C richtig zu erlernen? Die Aufgaben wie "clash" könnten immer noch als freiwillige Bonusprojekte machen.

Die Namensliste, bei der jeder unterschreibt, ist aus Datenschutzperspektive bedenklich. Es wäre möglich, die Liste heimlich mit dem Handy abzufotografieren, ohne dass es jemand bemerken würde, und so die Namen und Unterschriften aller Studierenden zu erfassen. Das ist wirklich problematisch.

Warum wird Waffel für die Verwaltung der Punkte verwendet und nicht StudOn, das optisch und von der Performance viel ansprechender ist?

- Die mach-Aufgabenstellung hat meiner Meinung nach nicht gut genug erklärt wie die Warteschlange zu nutzen war
- Die Übungen, in denen Stoff besprochen wurden, waren oft sehr komprimiert. Es war wenig Zeit zum Mitdenken vorhanden. Die Besprechungen hingegen dauerten oft nur 20 Minuten. Hier wäre eine geschicktere Aufteilung gut. Z.B. eine Übung, in der der erste Teil des Stoffes besprochen wird, und eine weitere, in der eine Aufgabe besprochen wird und der Rest der inhaltlichen Übung.
- Die Übungsaufgaben in SP sind einfach viel zu umfangreich und kompliziert für Studenten die gerade erst C Programmieren lernen. Man wird durch Vorlesung und Übung nicht gut genug auf so komplizierte Aufgaben vorbereitet und selbst wenn man ein Programm schreibt welches die Funktion erfüllt bekommt man kaum Punkte weil es für Fehlerbehandlung oder komische Randfälle Abzug gibt
- Es wäre schön, wenn die Anmeldung zu den Übungen wie bei den meisten anderen Veranstaltungen auf studon wäre.
- Evtl. Einheitlichere Abgabetermine, war teils etwas verwirrend
- Für Nicht-Informatiker mit entsprechend fehlenden theoretischen Vorkenntnissen zur Systemprogrammierung waren das Verstehen und der Zweck der Programmieraufgaben oft schwer nachvollziehbar und sehr abstrakt. Der Mehrwert der Aufgaben und deren Inhalte für Ingenieure und technische Angelegenheiten wurde nicht sehr deutlich.
- Ich finde das Prinzip sehr fraglich so viele wie möglich aus der Übung zu ekeln. Indem die Aufgaben von Anfang an sehr groß sind und die Korrektur so hart ist, das man nach 4 Fehlern keine Punkte mehr hat. Nicht sehr motivierend. Eine Vergabe nach dem was gepasst hat fände ich sinnvoller.
- Manchmal schadet es bestimmt sich nicht in die Übungsleitervorbereitung zu gehen ;)
- Stoff war teilweise für das zweite Semester sehr anspruchsvoll.
- Uebungen dauerten viel zu lange zu bearbeiten (bei mir pro Aufgabe 6+ Stunden) Die Aufgaben am Ende der Uebung waren fuer mich kaum zu bearbeiten ich haette deutlich mehr Zeit gebraucht um den Stoff der Uebung nochmal durchzugehen waehrend andere bereits nach wenigen Minuten die Loesung diktieren
- Wieso kein StudOn Kurs, sondern eine eigene Webseite? So vergisst man häufig, dass man in SP noch was zu tun hat, da der Kurs einfach nicht in der Übersicht erscheint. Alleine ein Kurs mit lediglich einer Verlinkung zur Webseite wäre eine Verbesserung.

Wieso Waffel? Kann man das nicht durch Studon ersetzen? Jede der Funktionalitäten ist dort auch gegeben.

Bzgl. Git: Eine Möglichkeit, seine Repos zu löschen wäre eine Erleichterung, da besonders bei Teamerstellungen mal Fehler passieren können.

7.8) Sonstiges:

- Da ich bereist vor 2 Jahren SPiC gehört habe und dieses Semester zum zweiten Mal SPiC-Tutor bin, fällt mir die LV entsprechend leicht.

- Der Sinn der Unterschriftenlisten in den Übungen, in denen nichts besprochen wurde, hat sich mir nicht erschlossen. Außerdem wäre ein Hinweis zum Vorgehen bei Krankmeldungen etc. wünschenswert.
- Die Bearbeitung der Programmieraufgaben in Gruppen über Gitlab hat größtenteils problemlos funktioniert
- Im aktuellen Format hat die Übung keine Darseinsberechtigung und ist didaktisch nicht sinnvoll. Umstrukturierung nötig.
- Vielleicht war die Änderung, mit der SP ab dem zweiten Semester bereits behandelt wird, keine Sinnvolle



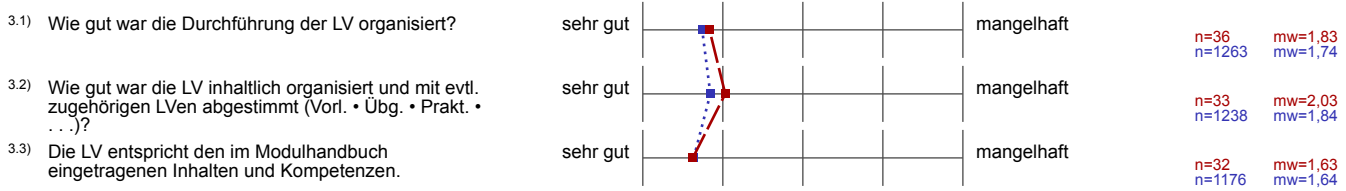
# Profillinie

Teilbereich: TF • Virtual-Class-Umfragen  
 Name der/des Lehrenden: Dr.-Ing. Jürgen Kleinöder  
 Titel der Lehrveranstaltung: Systemprogrammierung 1 - Übung  
 (Name der Umfrage)

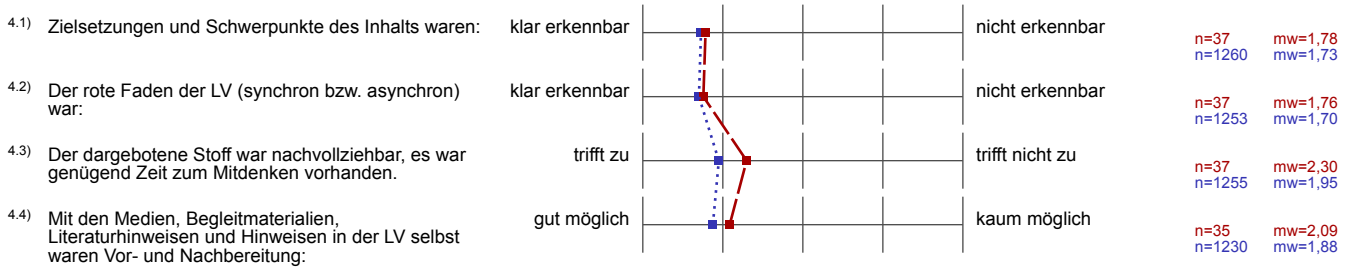
Vergleichslinie: Mittelwert-aller-Übungs-Rückläufer\_SS-23

Verwendete Werte in der Profillinie: Mittelwert

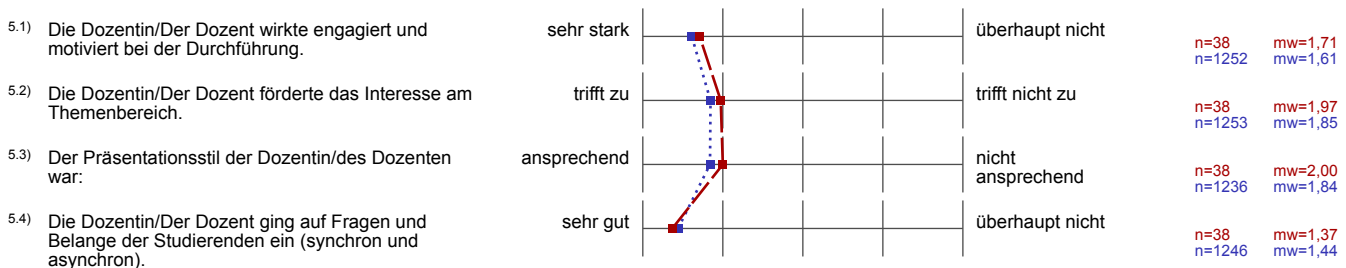
## 3. Organisation, Inhalte und Kompetenzen der Lehrveranstaltung



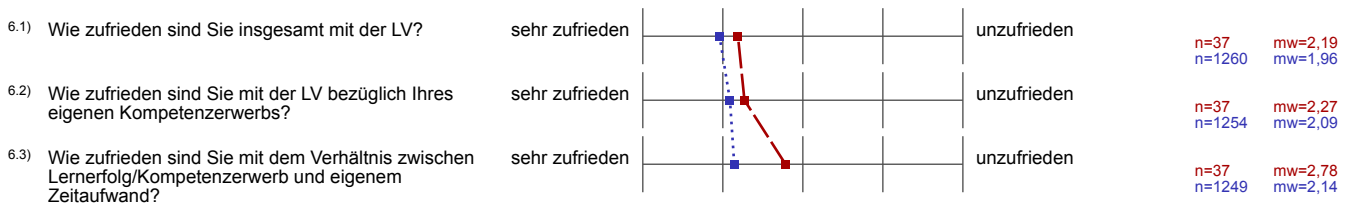
## 4. Struktur der Lehrveranstaltung



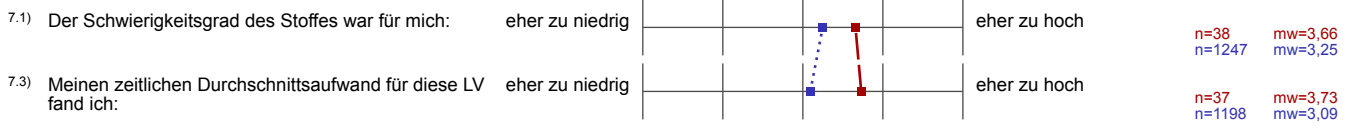
## 5. Durchführung der Lehrveranstaltung



## 6. Zufriedenheit und Kompetenzerwerb



7. Zusätzliche Informationen für die Dozentin/den Dozenten



# Profillinie für Indikatoren

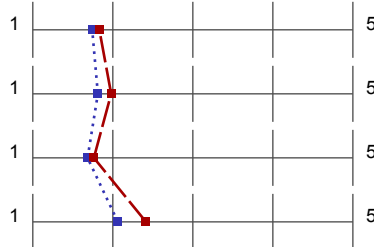
Teilbereich: TF • Virtual-Class-Umfragen

Name der/des Lehrenden: Dr.-Ing. Jürgen Kleinöder

Titel der Lehrveranstaltung: Systemprogrammierung 1 - Übung  
(Name der Umfrage)

Vergleichslinie: Mittelwert-aller-Übungs-Rückläufer\_SS-23

Indikator • Organisation, Inhalte und Kompetenzen der LV (Kap. 3)



mw=1,83 s=0,95  
mw=1,74 s=0,95

Indikator • Struktur der LV (Kap. 4)

mw=1,98 s=1,02  
mw=1,81 s=1,00

Indikator • Durchführung der LV (Kap. 5)

mw=1,76 s=1,10  
mw=1,68 s=1,00

Indikator • Zufriedenheit und Kompetenzerwerb (Kap. 6)

mw=2,41 s=1,34  
mw=2,06 s=1,12