



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

Sehr geehrter Herr
Dipl.-Inf. Michael Stilkerich (PERSÖNLICH)

SS 2008: Auswertung für Übungen zu Systemnahe Programmierung in C

Sehr geehrter Herr Dipl.-Inf. Stilkerich,

Sie erhalten hier die Ergebnisse der automatisierten Auswertung der Lehrveranstaltungsevaluation im SS 2008 zu Ihrer Umfrage:

- Übungen zu Systemnahe Programmierung in C -

Es wurde hierbei der Fragebogen - ü_s08 - verwendet, es wurden 84 Fragebögen von Studierenden ausgefüllt.

Die Note 1 kennzeichnet hierbei eine maximale Güte, die Note 6 eine minimale Güte für die einzelnen Fragen bzw. Mittelwerte.

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

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.

Es folgen die von Ihnen evtl. selbst gestellten Fragen, 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://eva.uni-erlangen.de> (--> Technische Fakultät --> Ergebnisse --> SS 2008) möglich, hierzu die Bestenlisten, etc. einsehen.

Bitte melden Sie an eva@techfak.uni-erlangen.de die Anzahl der ausgegebenen TANn, wenn Sie das bis jetzt versäumt haben.

Mit freundlichen Grüßen

Bernhard Schmauß (Studiendekan, bernhard.schmauss@lhft.eei.uni-erlangen.de)
Jürgen Fricke (Evaluationskoordinator, eva@techfak.uni-erlangen.de)

Dipl.-Inf. Michael Stilkerich

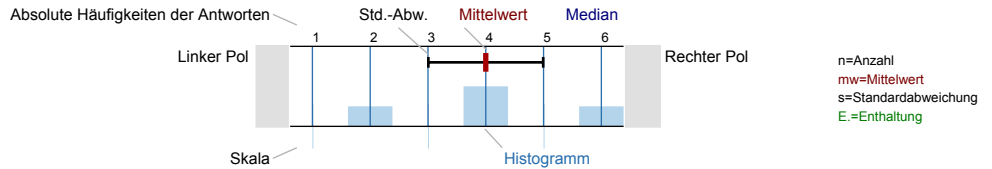
Übungen zu Systemnahe Programmierung in C (08s-Ü SPiC)
Erfasste Fragebögen = 84

Globalwerte

Globalindikator		<b style="color: red;">mw=2.27 s=1.21
Übung im Allgemeinen		<b style="color: red;">mw=2.23 s=1.17
Didaktische Aufbereitung		<b style="color: red;">mw=2.43 s=1.25
Persönliches Auftreten des Übungsleiters		<b style="color: red;">mw=2.23 s=1.32
Verwendete Hilfsmittel		<b style="color: red;">mw=2.17 s=1.1
Gesamteindruck		<b style="color: red;">mw=2.28 s=1.21

Legende

Frage text



Achtung: Beim Anklicken der Sprachsymbole verlieren Sie alle bisherigen Eintragungen !!

Allgemeines zur Person

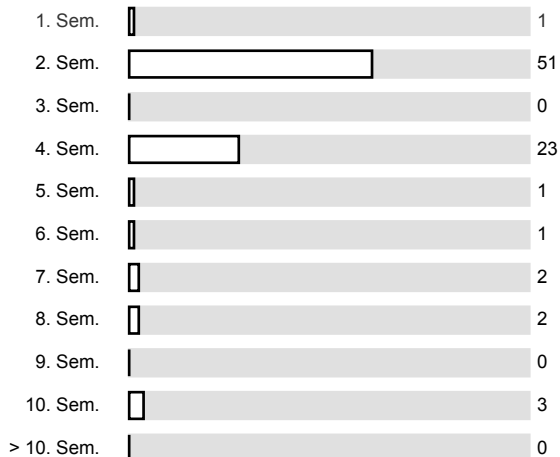
^{2_A)} Ich studiere folgenden Studiengang:

n=84

EEI B.Sc.	<input style="width: 100px; height: 15px;" type="text"/>	37
EEI Dipl.	<input style="width: 100px; height: 15px;" type="text"/>	1
EI Lehramt	<input style="width: 100px; height: 15px;" type="text"/>	1
MB B.Sc.	<input style="width: 100px; height: 15px;" type="text"/>	2
MECH B.Sc.	<input style="width: 100px; height: 15px;" type="text"/>	11
MECH Dipl.	<input style="width: 100px; height: 15px;" type="text"/>	29
(T)Math B.Sc.	<input style="width: 100px; height: 15px;" type="text"/>	2
Sonstiges	<input style="width: 100px; height: 15px;" type="text"/>	1

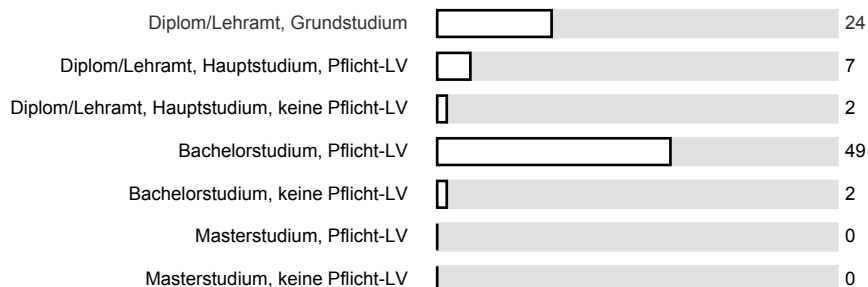
2_B) Ich bin im . . . Fachsemester.

n=84



2_C) Diese Lehrveranstaltung gehört für mich zum . . .

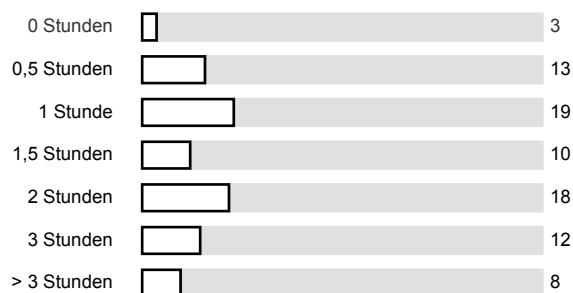
n=84



Mein eigener Aufwand

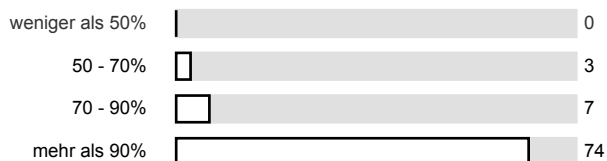
3_A) Mein Durchschnittsaufwand für Vor- und Nachbereitung dieser Übung beträgt pro Stunde (45 Min.):

n=83



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

n=84



Übung im Allgemeinen

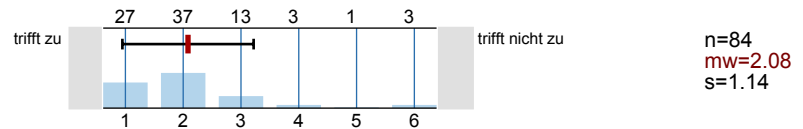
4_A) Zielsetzungen, Struktur und Schwerpunkte des Übungsinhalts sind:



4_B) Umfang und Schwierigkeit der Übung ist:

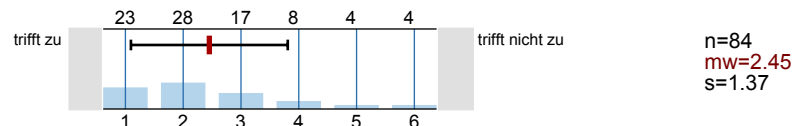


4_C) Die Übungsform (Aufgabenbehandlung, Programmieren, etc.) ist gut zur Vermittlung des Stoffes geeignet.

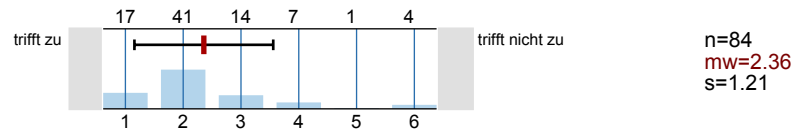


Didaktische Aufbereitung

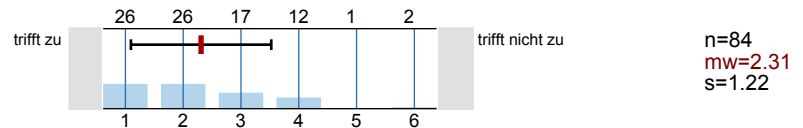
5_A) Die Präsentation von Aufgaben und Lösungen ist nachvollziehbar, es ist genügend Zeit zum Mitdenken vorhanden.



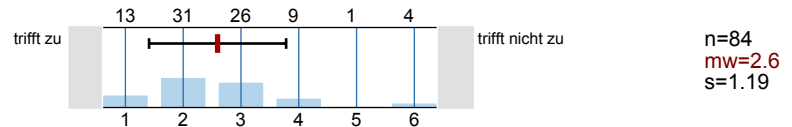
5_B) Die Anwendbarkeit des Übungsstoffes wird durch Beispiele gut verdeutlicht.



5_C) Ich werde gut zum selbstständigen Lösen von Aufgaben angeleitet.

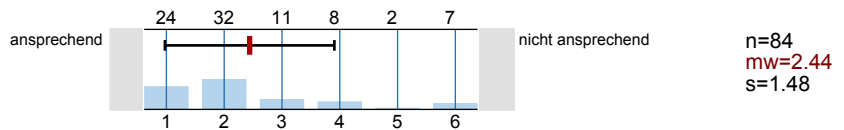


5_D) Der Bezug zur Vorlesung und den Prüfungsanforderungen wird hergestellt.



Persönliches Auftreten des Übungsleiters

6_A) Der Präsentationsstil des Übungsleiters ist:

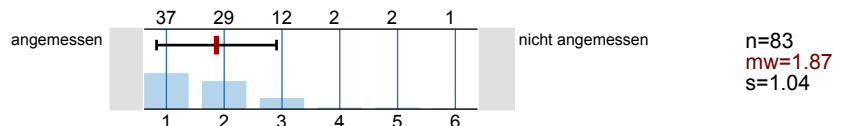


6_B) Der Übungsleiter vergewissert sich, dass die Problemstellungen und Lösungen verstanden wurden und geht gut auf Zwischenfragen ein.

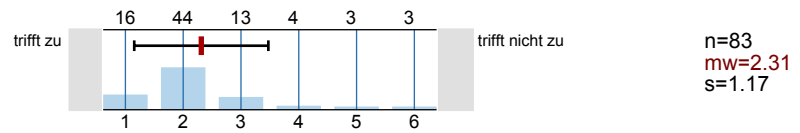


Verwendete Hilfsmittel

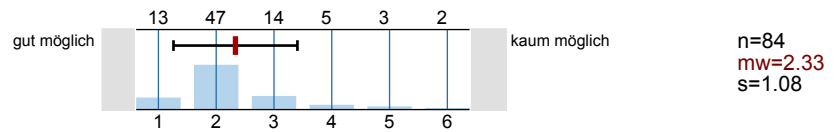
7_A) Der Einsatz von Medien (Tafel, Overhead-Projektor, Beamer, etc.) ist:



7_B) Die zur Verfügung gestellten Unterlagen sind in Menge und Qualität den Zielen der Übung angemessen.

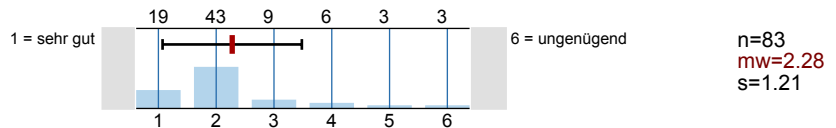


7_C) An Hand des erarbeiteten Materials ist die Vertiefung des Vorlesungsinhalts:



Gesamteindruck

8_A) Insgesamt lautet mein Urteil (Note 1 bis 6) für diese Übung:



Weitere Kommentare

9_A) An der Lehrveranstaltung gefällt mir besonders:

- Das System der Bonuspunkte bringt dem Dozenten/den Übungsleitern sehr viel, dadurch dass viele Studenten wirklich interessiert an einer perfekten Abgabe sind und auch dementsprechend Zeit und Mühe aufwenden - unbedingt beibehalten und weiterempfehlen!!
- Dozent hat sich sehr bemüht, den Vorlesungsstoff zu vermitteln und auch auf die einzelnen Probleme der Studenten (vgl Klausur-Termin der Dipl.Studenten)einzugehen
- kleine Gruppen
- Sicherha war ein ganz cooler Uebungsleiter, auch wenn er kein STudi hat. Ist gut auf Probleme eingegangen, konnte immer helfen!
- Uebungsleiter war stets gut vorbereitet. Endlich mal praxisbezogene Programmierung (nicht wie in Java).
- Vorstellung alternativer Loesungen; Praesentationsstil des Uebungsleiters;
- Bei den letzten Übungen hat man besser praktisch sehen können, was man da eigentlich programmiert hat, auch wenn diese aufgaben vielleicht etwas schwerer waren.
- Die Caro macht's einfach gut.
- Ich besuche die Übung Mittwoch um 10:00 bei Christoph Erhardt. Der Übungsleiter hat sich intensive um das Verständnis bei den Studenten bemüht. Er hat sich sehr viel Zeit genommen und ist sogar länger als seine eigentliche Rechnerübungszeit im Raum geblieben. Er hat sich versucht in die Probleme der Studenten hinein zu versetzen. Alles in allem war der Besuch der Übung sehr lohnenswert. Eine der besten Übungen die ich je besucht habe, obwohl ich sonst keinen großen Gefallen an Informatik habe. Die 10% Klausurbonus waren natürlich eine große Antriebskraft und sind meiner Meinung nach auch sehr gut angekommen. Weiter so.
- Praxisnah
- Studenten werden zum selber programmieren angeregt.
- Praxisbezug duch die µC-Boards ist vorhanden. Man kann "sehen" was man programmiert hat. Gut war auch, dass sehr schnell mehr Server eingerichtet wurden, und die nicht einfach gesagt wurde "Naja, müsst ihr halt mal länger warten bis ihr eure Programme starten könnt"
- dass man die schwierigeren Sachen schon einmal in der Übung macht und nicht wie in GDI 1 8 Stunden vor dem Rechner sitzt und total verzweifelt! das programmieren auf der echten Hardware hat wirklich Spass gemacht!
- In den Rechnerübungen waren die Tutoren sehr nett und hilfreich. Man wurde gut an die Lösung herangeführt, ohne das einem ein Weg "vorgeschrieben"wurde.
- Viele Rechnerübungen und Übungsleiter, die einem was erklären können. Rücksichtsvoller Tafelübungsleiter

- Schließt sehr gut an Gdl 1 (Java Programmierung) an und macht die Unterschiede der Programmiersprachen deutlich.
- - Nicht nur C, sondern auch praxisnahe AVR-Programmierung inkl. Board, ein paar Boards mehr wären allerdings wünschenswert; trotzdem: super!

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

- betrifft Tafelübung 5 bei Hackner: - Raum ist eine absolute Zumutung, wurde seit Semesterbeginn oder länger nicht gereinigt (Kaffeeflecken neben Bildschirm bereits vor erster Übung da und werden auch nach der letzten nicht beseitigt sein; Tastaturen kleben, Maus hat "erhöhten" Reibungswiderstand auf Tischfläche wegen Dreck --> RICHTIG EKLIG) - Recherausstattung unpraktisch, Tastaturen mit englischen Layouts (wir leben in Deutschland!!!) machen das Arbeiten schwierig, zusätzlich sind auch noch an verschiedenen Rechner unterschiedliche englische Tastaturen angeschlossen mit unterschiedlicher Belegung. Effektives Arbeiten wird dadurch verhindert. - etwas blöderes, wie den einzigen unter Linux lauffähigen Compiler für Microcontroller unter Windows über Remotedesktop zu verwenden, ist scheinbar keinem mehr eingefallen: Ressourcenverschwendung pur (die zusätzlichen Rechner die im Hintergrund laufen kosten zu 100% mehr als die paar Campus-Lizenzen von Windows. Aber warum einfach, wenns auch umständlich geht), kein Zugriff auf Aufgabenstellungen unter Windows erlaubt, zur Verfügung gestelltes Tool HAPSIM über Remotedesktop praktisch nicht anwendbar, da entweder AVR-Simulator läuft und HAPSIM nicht reagiert oder anderstherum - bis jetzt sind immer noch nicht die Punkte letzten korrigierten Aufgabe eingetragen (3 Wochen überfällig)
- Zu wenig "Ampel-Boards". Die Materialkosten belaufen sich auf 5€ pro Stück und könnten im Zuge eines Seminars an der Uni erstellt werden. So hätte jeder die Möglichkeit, sein Programm gleich ohne das lästige Heapsim zu testen!
- In Rechnerübungen sollten 2 Tutoren dauerhaft anwesend sein, da ein Tutor durch den Andrang häufig überwältigt war und lange Wartezeiten entstanden.
- Übungsleiter blaettert Folien im Schnelldurchgang durch, kann sehr schlecht den Übungsstoff erklären
- Für den Umfang der Hausaufgaben sind die Übungsfolien und die Vorlesungsfolien viel zu wenig. Vieles schreibt man dann nur ab, ohne es wirklich kapiert zu haben was man eigentlich gerade macht. Die Übungsfolien sind viel zu Stichpunktartig.
- Sun-Tastatur
- Dozent sollte langsamer sprechen, und evtl nach 45 min eine ca 4-5 min Pause einlegen :-)
- teilweise zu schwere Aufgabenstellung, zu volle Rechnerübungen
- Zu wenig eigenes denken...Mehr abschreibeaufgabe... Aufgabenstellung war dann wiederum nicht eindeutig, verschieden Übungsleiter wollten andere Lösungsansätze -> Punktabzug bei Kleinigkeiten...
- Jeder Student sollte - finanziert durch Semesterbeiträge - einen AVR Bausatz bekommen und als 0-te Aufgabe zusammenschweißen. Zwei Boards auf so viele Studenten sind eindeutig zu wenig. Manche Übungstermine haben sich überschritten mit Etech Praktikum. Die ersten drei Aufgaben waren etwas zu einfach. Für die schweren Aufgaben war die Bearbeitungszeit zu kurz.
- evtl mehr Beispiele mit ausführlichen Kommentaren (auch zu ähnlichen/verwandten Aufgaben) um weiteren Einblick in Funktionen/Anwendungsfälle zu bekommen; die Beispiele müssen nicht unbedingt mit der Übung zu tun haben, aber sie sollten das Transferdenken "aktivieren"
- Ich finde es nicht gut, dass der Übungsgruppenleiter sogenannte "Interviews" durchführt, um zu überprüfen, ob die Teilnehmer ihre Aufgaben selbst bearbeitet haben. Mit einem Interview pro Semester kann der Tutor das sowieso nicht rausfinden.
- Jens Schedel auswechseln. Punktabzüge auf teilweise triviale Sachen, die ausschließlich seiner persönlichen Präferenz entsprechen (Jede Deklaration in eine eigene Zeile zum Beispiel), Abfrage der ersten Übungsaufgabe auch noch nach 6 Wochen?!
- Die Übungsaufgaben waren am Ende schon recht anspruchsvoll, so dass wir eigentlich dachten wenn wir diese Aufgabe verstehen, verstehen wir die Klausuren auch besser. Man versteht jetzt zwar, was man in der 2. Aufgabe der Klausur von einem will, allerdings sind die Klausuraufgaben noch deutlich anspruchsvoller als die Übungen, was uns erst einmal sehr erschrocken hat.
- etwas realitätsferne Programmierung... in der Praxis macht man einige Sachen anders. Viele verwirrende Sachen.
- Hapsim hat nicht so funktioniert wie die Hardware.
- Hapsim funktioniert nicht richtig!!

- Vielleicht sollte der Übungsleiter Christoph Erhardt sich in der Rechnerübung etwas weniger Zeit für die Einzelnen nehmen, da es manchmal schon sehr lange gedauert hat bis man seine eigentlich sehr kurze Frage dann stellen konnte. Das ist natürlich sehr gut, wenn man etwas ausführlich erklärt bekommt. Auf der anderen Seite müssen die anderen Studenten sehr lange warten. Vielleicht wäre es besser einzelne Studenten kürzer, dafür öfter zu beraten.
- Organisatorisch etwas verwirrend
- Übungsleiter (Michael) reden oft zu schnell und auf einem viel zu hohen Niveau (Fachjargon). Für Anfänger sind viele Dinge nicht so selbstverständlich wie für einen geübten Programmierer.
- Musterlösungen waren sehr hilfreich. Besonders bei der letzten Aufgabe. Unser Übungsleiter konnte uns zwar erklären, was wir hätten anders machen können, aber dies in schriftlicher Form wäre noch besser gewesen.
- In den Rechnerübungen musste man leider teilweise sehr lange warten, bis ein Übungsleiter Zeit hatte. Das sollte verbessert werden. Manchmal war der Stoff auch viel zu viel und vor allem zu schnell in der Präsentation (also den Folien der Übung) mit zu wenig Beispielen zur Verdeutlichung was der Quelltext jetzt eigentlich bewirkt. Die Aufgabenstellungen sollten exakter formuliert werden, um Missverständnisse zu vermeiden.
- Ich habe vermisst, Programme mit mehreren .c-Dateien und Makefiles usw zu entwickeln (2SWS).
- Die Aufgabenstellungen waren häufig ziemlich schwammig, keiner wusste genau welche Ausnahmefälle (falls man überhaupt selber auf diese kommen konnte oder dass man sie gesagt bekam) wie und warum behandelt werden sollen. Auch waren die Folien für die Übung sehr schwach, da gute Beispiele fehlten. Häufig stand zwar etwas beispielhaft da, aber meist nur eins, sodass es schwer war rauszufinden wie man es denn anders machen kann. Zudem war es häufig nicht klar, wo im Programmtext denn nun zB Timerbefehle usw gesetzt werden mussten. Außerdem wären Musterlösungen für die Programmieraufgaben nützlich, damit man besser nachvollziehen kann warum und wofür man was abgezogen bekommen hat bzw sich ein Bild davon machen kann, wie das Programm ideal funktioniert hätte.
- das Übungsprogramm Hapsim sollte einmal ordentlich überarbeitet werden. Das z.B. der Taster wenn man das geschriebene Programm laufen lässt am Anfang gedrückt ist kann einfach nicht sein! und dass die Taster in der Simulation funktionieren, obwohl man sie im Programm nie einschaltet ist auch etwas seltsam
- Wenn die Lösungen vorgestellt werden sollen, dann sollte auch jeder in der Gruppe mal drankommen und das dann auch in allen Gruppen!
- trotzdem wurde manchmal zu viel vorausgesetzt
- -die Studenten den Code/Strukturen öfter selbst erarbeiten lassen
- Schließt sehr gut an Gdl 1 (Java Programmierung) an und macht die Unterschiede der Programmiersprachen deutlich.
- - Der Beamer im Tafelübungsraum (glaube es war 01.155-N) war leider mehr oder weniger kaputt (türkis-Stich oder so), jedenfalls erkennt man ab der 3. Reihe nichts mehr auf der Leinwand... Ein neuer Beamer müsste her. - Die Übung als Pflichtveranstaltung zu machen ist, wie ich finde, nicht "studenten-gemäß" in dem Sinne, dass (programmier)erfahrenere Studenten einfach daheimbleiben könnten, anstatt sich in der Übung zu langweilen - Die Idee, Bonuspunkte für die Klausur zu vergeben, die von der Anzahl der erreichten Hausaufgaben-Punkte abhängen, ist an sich nicht schlecht. Allerdings schade, dass man, um die Punkte der Hausübung zu bekommen (auch wenn die Hausaufgabe vollkommen korrekt gelöst wurde!), in der Tafelübung anwesend sein muss. - Die Hausübungen sind vom Schwierigkeitsgrad her völlig unausgewogen. Die ersten Programme waren zu leicht (bis swap); der Grund hierfür ist schlichtweg, dass die Lösung der Aufgaben zu großen Teilen quasi aus copy'n'paste der Folien der VL oder der Übung bestand.. Das war teilweise auch noch beim Atmega8 so. Die vorletzte Aufgabe allerdings (mini_sh) fanden viele Studenten so schwer, dass sie die Lösung bis zu jetzt nicht hinbekommen haben; das lag wohl daran, dass sowohl die VL-Folien als auch die der Übung nicht ausführlich genug für die Themen Prozesse und Signale unter UNIX waren. Hier musste man, wenn man die Aufgaben lösen wollte, ausführlichere Literatur zu Rate ziehen (was ja nicht schlecht ist!). Am besten wäre es jedoch, alle Aufgaben so wie die letzte Aufgabe zu gestalten. Hier wurde in der Übung eine gute "Anleitung" zur Vorgehensweise und Hinweise auf die "knackigen" Teile der Aufgabe gegeben (z.B. Umgang mit qsort (pointer auf pointer!)). Trotzdem musste man schon auch selber noch nachdenken. Am besten, alle Aufgaben so gestalten, dann bringen die Hausübungen auch richtig was!
- Die Rechnerübungen sollten von mehr Übungsleitern betreut werden.

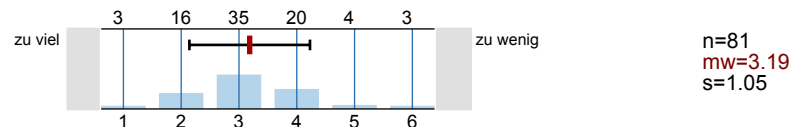
^{9-C)} Zur Lehrveranstaltung möchte ich im Übrigen anmerken:

- Eventuell präziser formulierte Aufgabenstellung... öfter kam es bei uns zu kleineren Missverständnissen! Jedoch wurden diese sofort auf Nachfrage per Email beantwortet. :-)
- Die Rechenübungen wurden zwar kompetent betreut, aber die Anzahl der Tutoren war nicht immer ausreichend.
- Ein Tutor ist in den Rechnerübungen zu wenig, der kann nicht alle frsagen ausführlich beantworten da zu viele Fragen von verschiedenen Studenten auftreten.

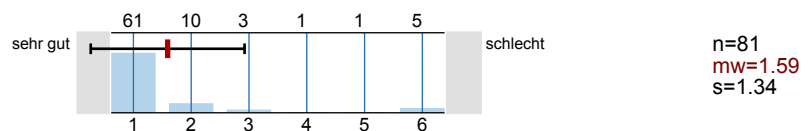
- Programmierung auf den AVR Prozessor war sehr praxisnah
- "Punktesystem" bei den Uebeungen war eine gute Idee
- Passt schon alles! Mehr eigenstaendiges denken waere nett...
- Die Bonuspunkte sind gut und vor allem die Moeglichkeit die Uebung zu zweit abgeben zu koennen, da relativ viel Zeit investiert werden muss und man die nicht immer unbedingt hat.
- Jens Schedel: Zu viel Abfragen, zu viel verträdelte Zeit - würde man nicht während der Besprechung und Abfrage programmieren, wäre eine Bearbeitung der Aufgaben in den Übungen nicht möglich.
- Bei manchen Rechnerübungen sollten dringend 2-3 Übungsleiter da sein. Da sonst der Übungsleiter irgendwann durchdreht und schon Listen führen muss, wer als nächstes dran kommt und die hilflosen Studenten manchmal eine halbe stunde warten müssen bis sie Hilfe bekommen.
- mehr avr in den kommenden semestern machen.
- Es ist schwierig ein komplettes Programm zu schreiben ohne jemals ein ganzes gesehen zu haben. Die die benötigten Programsegmente sind zwar vollständig, aber die einordnung ins ganze ist anfangs sehr schwer

Optionale Zusatzfragen des Übungsleiters

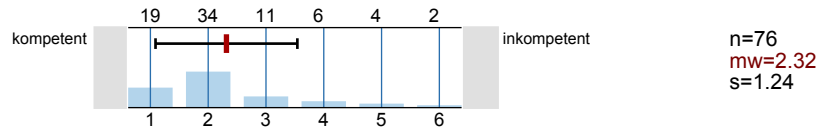
10_A) Übungsaufgaben



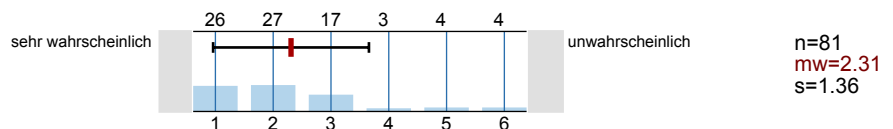
10_B) Was halten Sie von dem Konzept der Bonuspunkte



10_C) Wie war die Betreuung in den Rechnerübungen



10_D) Erwarten Sie den erlernten Stoff im späteren Berufsleben oder privaten Projekten wiederverwenden zu können



10_K) Spezielle Anmerkungen zu Ihrem Übungsleiter (bitte Namen des Übungsleiters oder Übungsnummer mit angeben)

- Jens Schedel: sehr gut auch um schlechte Informatiker gekümmert, hilfsbereit, Zusatz"nachhilfe"
- Jens Schedel: sehr hilfsbereit, setzt sich stark ein,damit die Aufgaben verstanden werden
- Christoph Erhardt
- Tafelübung 12
- Netter und hilfsbereiter Uebungsgruppenleiter! (Tobias)
- Christoph Erhardt ist sehr gut auf unsere Fragen eingegangen und konnte den Stoff meist gut an den Mann/die Frau bringen
- Sicherha war immer fuer uns da
- Stilkerich: kompetent, vermittelt den stoff gut, geht auch gut auf zwischenfragen ein
- Schedel denkt er is super cool.
- Tobias: War schon hilfsbereit, hat nur die Folien der Übung manchmal etwas schnell runtergerattert ;)

- bei der caro: flotte dame, sehr kompetent. danke für die übung.
- Vince war cool
- Übungsleiter (Michael) reden oft zu schnell und auf einem viel zu hohen Niveau (Fachjargon). Für Anfänger sind viele Dinge nicht so selbstverständlich wie für einen geübten Programmierer.
- Jens Schedel: Scheint das Konzept des "Nachvollziehens am Rechner während der Übung" nicht ganz verstanden zu haben. Redet 90 Minuten durch, so dass man gar keine Chance hat, irgendetwas am Rechner nachzuvollziehen. Hat ausserdem die Angewohnheit, v.a. die Leute an die Tafel zu holen, die die Aufgaben falsch haben. Dass man auf die häufigsten Fehler eingeht ist wirklich gut, aber man sollte es den Leuten vielleicht doch ersparen, sich vorne an der Tafel verantworten zu müssen. Ein wenig brauchbarere Hilfestellungen/Tipps, die man zum Programmieren braucht, wären auch mal ganz nett gewesen...
- Caro hat die Aufgaben leider sehr langsam verbessert und zurückgegeben. Mit Wartezeiten bis 4 Wochen kann man die Verbesserungen nicht in die nächste Aufgabe mit einfließen lassen!
- unsere Übungsleiterin (Caro) hat sehr lange zum korrigieren gebraucht, wir haben die Verbesserung von laufflicht.c erst bekommen, als die 2. ampel.c schon fast abgegeben sein musste!
- Übungsgruppe 07 kompetenter hilfsbereiter Übungsleiter
- Erwarten Sie den erlernten Stoff im späteren Berufsleben oder privaten Projekten wiederverwenden zu können: - Im Praktikum Mechatronische Systeme schon geschehen. - Beim Hobby Elektronik wird es noch geschehen

Profillinie

Teilbereich:	Technische Fakultät (TF)
Name der/des Lehrenden:	Dipl.-Inf. Michael Stilkerich
Titel der Lehrveranstaltung: (Name der Umfrage)	Übungen zu Systemnahe Programmierung in C
Vergleichslinie:	Alle Übungs-Fragebögen im SS08

