



^b
**UNIVERSITÄT
BERN**

Vizerektorat Qualität, Fachstelle Lehrevaluation, Hochschulstrasse 6, 3012 Bern

Persönlich/Vertraulich
Mr.
Prof. Dr. Oscar Marius Nierstrasz
Institut für Informatik und angewandte Mathematik
Neubrückstr. 10
3012 Bern

Vizerektorat Qualität
Fachstelle Lehrevaluation

Report of evaluation: FS16 Programming Languages (2720)

Dear Prof. Dr. Nierstrasz

Please find here the results of the evaluation of your course Programming Languages. Following the scanning of the questionnaires, this report was automatically generated and mailed to you.

The questionnaire used was PN-P1.V1. In the report, you first see the mean values of the following dimensions:

Planning and Presentation (Skalenbreite: 4)
Manners with Students (Skalenbreite: 4)
Interest and Relevance (Skalenbreite: 4)
Complexity and Scope (Skalenbreite: 5)
Overall Assessment (Skalenbreite: 6)

In the second part of the report, you see the answers to all the questions. The number of answers, the mean value and the values differing from it are also given.

Grade 1 equals the lowest grade given by the students, grade 4 or more the highest grade (unless a question is reversed). In 'complexity and scope' grade 3 corresponds to 'exactly right' and is therefore the best grade. In the overall assessment of the Course, grade 6 means the best result.

We hope that this report helps you to analyse your course. Please briefly discuss the results with your students before the end of the semester.

In case you wish to learn more about how to improve your teaching, you might want to discuss the results with the staff of the 'Hochschuldidaktik' (mail address: hd@zuw.unibe.ch). Please bring a copy of the report with you, since the staff of Hochschuldidaktik do not have access to evaluation results.

You might find guidelines, regulations and information about the process under www.lehrveranstaltungsevaluation.unibe.ch (documents in German).

Should you need more information, you may also contact us by e-mail.

Kind regards
Daniela Wuillemin
Vice-rectorate of quality

Daniela Wuillemin
Vizerektorat Qualität
Evaluationsstelle
Hochschulstrasse 6
3012 Bern

Tel. +41 31 631 51 07
daniela.wuillemin@qualitaet.unibe.ch
www.lehrveranstaltungsevaluation.unibe.ch

Overall indicators

Planning and Presentation (Skalenbreite: 4)



Manners with Students (Skalenbreite: 4)



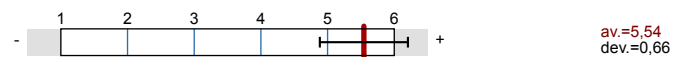
Interest and Relevance (Skalenbreite: 4)



Complexity and Scope (Skalenbreite: 5)

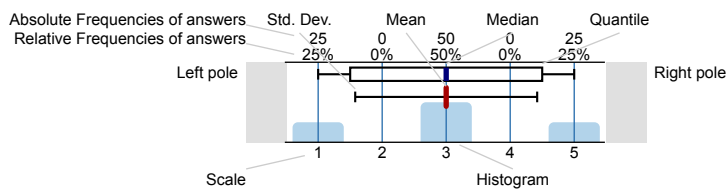


Overall Assessment (Skalenbreite: 6)



Legend

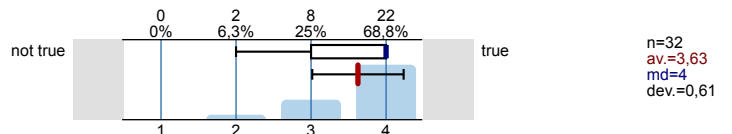
Question text



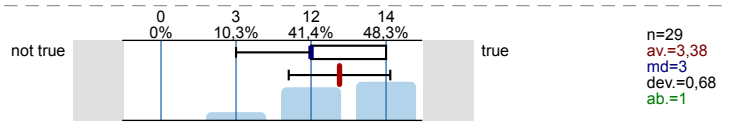
n=No. of responses
av.=Mean
md=Median
dev.=Std. Dev.
ab.=Abstention

Planning and Presentation

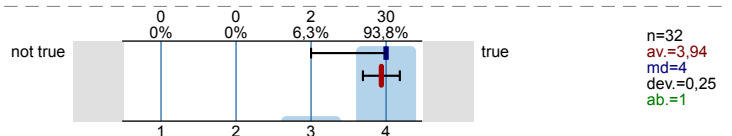
1 The course follows a coherent structure.



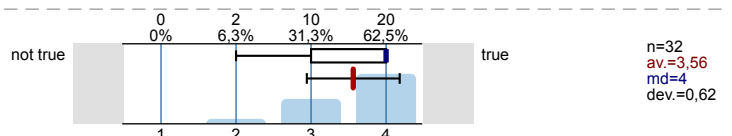
2 The wider context of the subject matter is sufficiently elucidated.



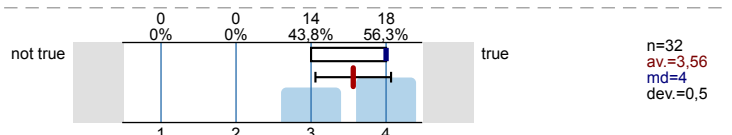
3 The lecturer expresses him-/herself clearly and comprehensibly.



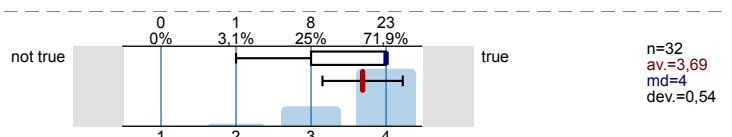
4 The course provides an adequate overview of the subject matter treated.



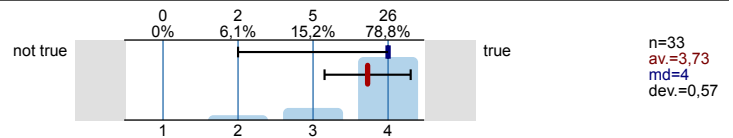
5 The design of the course contributes to an understanding of the subject matter.



6 There is overall enough material provided to assist the learning process (slides, course material, hand-outs, etc.).

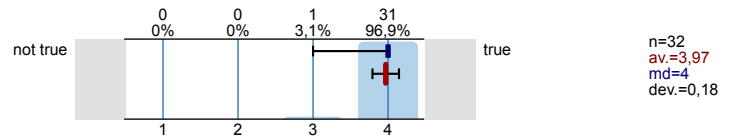


7 The course materials (slides, course manuals, hand-outs, etc.) are overall of sufficient quality.

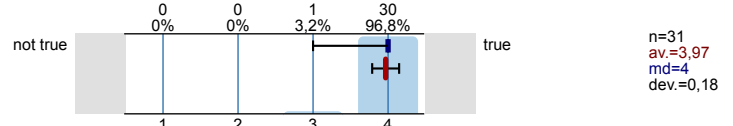


Manners with Students

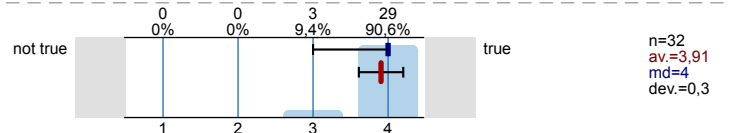
8 The lecturer takes students seriously.



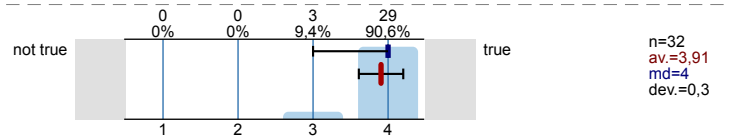
9 The lecturer is friendly and respectful towards students.



10 The lecturer addresses questions and suggestions from students adequately.

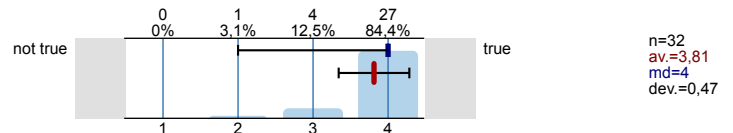


11 The lecturer seems to care about his/her students' progress.

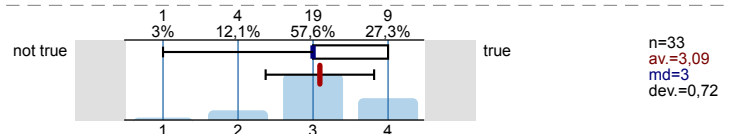


Interest and Relevance

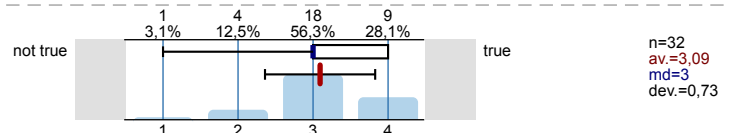
12 The lecturer succeeds in making the course interesting.



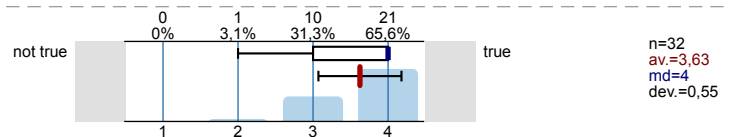
13 The course is probably very useful for my future professional life.



14 The applicability and relevance of the subject matter is sufficiently clarified by the lecturer.

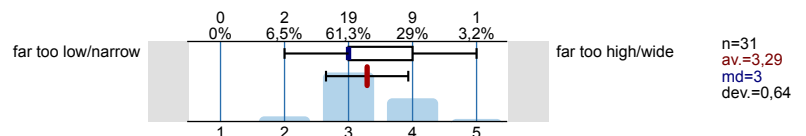


15 The lecturer fosters my interest in the subject.

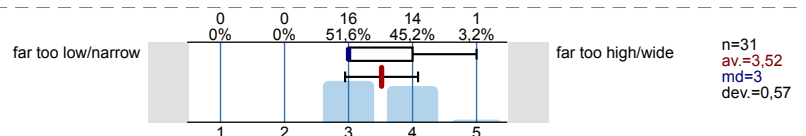


Complexity and Scope

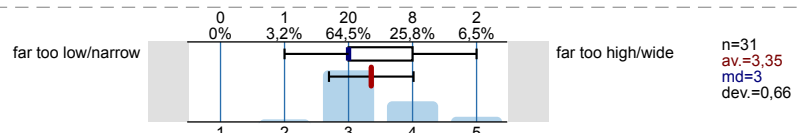
16 The degree of complexity of the course is:



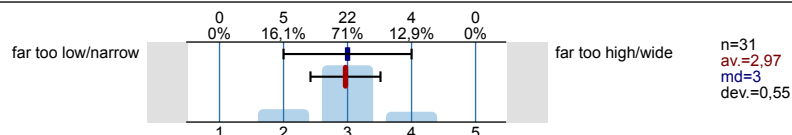
17 The scope of the course is:



18 The pace of the course is:

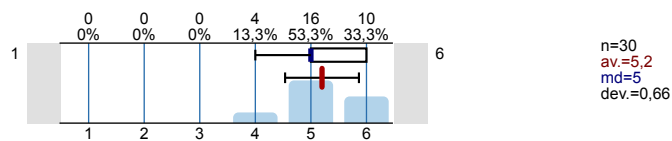


19 The amount of knowledge presupposed by the course is:

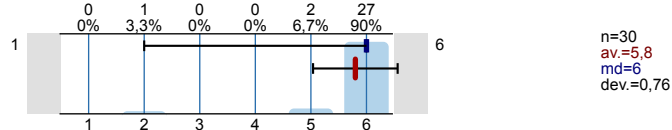


Overall Assessment

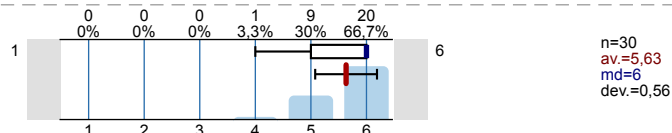
20 How would you grade the course as a whole?



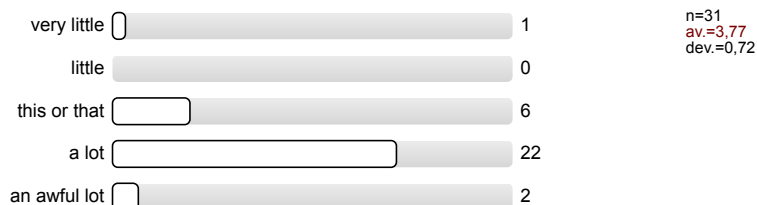
21 How would you grade the lecturer with regard to subject expertise?



22 How would you grade the lecturer with regard to teaching methods?

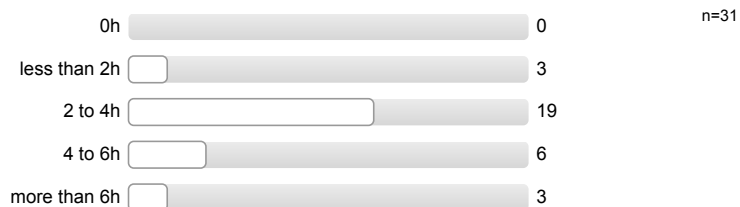


23 The course has taught me



Socio-demographic Data and Background Variables

24 How many hours per week did you invest in preparation and revision for the course (on average)?



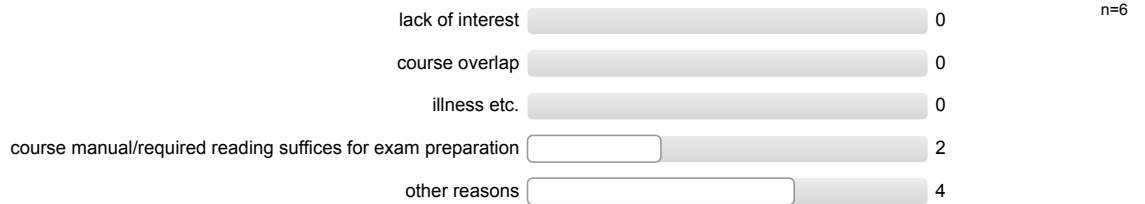
25 Was the topic of interest to you?



26 How many lectures did you miss?



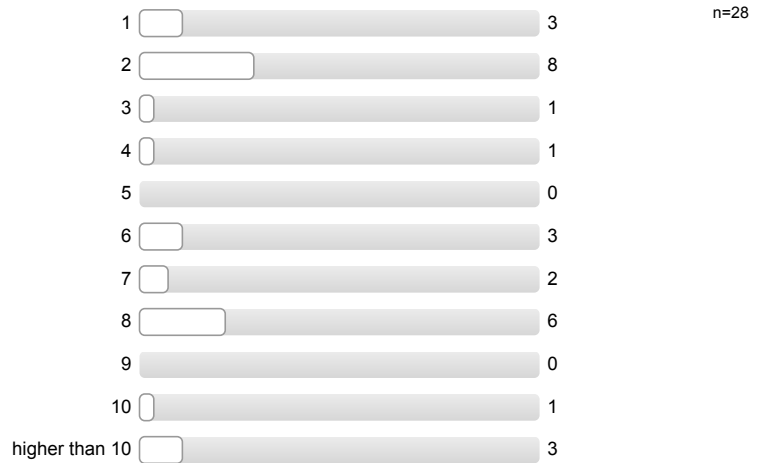
27 If you missed more than 2 lectures, please give one reason:



28 Allocation of the course in your study programme:?



29 Your current number of semesters?



30 Sex



Open Questions

What did you like about the course?

The wide range of subjects

Theory as well as practical exercises

Very good lectures, interesting topics, good examples.

Presentation of not (yet) so popular languages and underlying concepts.

good slides and presentation style

New aspect of PL
New PL learned

Broad overview over many topics

The practical exercises.
The understandable introduction to λ -calculus

I got a nice overview about programming languages
I love the practical demonstrations in class
Practical Exercises were often fun and interesting

The wide variety of programming paradigms improved my understanding of various language features. The lecturers were excellent and made sure the complex contents were understood well.

Everything: ~~to be~~ and I mean it.

Varied array of topics / languages

Introduction to new paradigms and languages

Practical exercises to get familiar with new programming languages

The mix between "theoretical" lectures (e.g. Types, Semantics) and "practical" lectures (e.g. Stack-based / PostScript, Prototypes / JavaScript).

The ~~draw~~ ~~draw~~ ~~over~~ overview was eye-opening

The notes in the slides are very helpful.

The way the professor explained some sometimes difficult subject in a very easy and interesting way.

As very well explained and hidden notes were a huge plus.

Content: Lots more to programming languages than what is popularly used.

What did you not like about the course?

theoretical parts (Types) and semantics

Subject is too broad, mostly scratching the surface of all paradigms.

The type lectures were a bit confusing, very, very hard to wrap my head around it. I have still not succeeded in that respect.

seems to have stopped at year 2000. Should probably be called "history of programming languages". What about Python, Go, Erlang. Assistants did nothing!

Some exercises were somehow too difficult others too easy. So I was sometimes really insecure, if I understood them right (when they were too easy)

Theoretical questions were often extracted from the slides and sometimes very hard to answer

Some concepts were a bit vague because of the high-level of the subject matter.

The exercises are not hard enough, and the course does not deserve the 5 ECTS in my opinion.

We got no feedback for the exercise (only partly solutions)

Exam date, there is too much material to be prepared for the time we had in disposition. And we might get bad grades, not that we didn't learn too much but we had no time to.

Some subjects were not very deep

The subject matter was enormous.

A lot of abstract meanings for its volume.

more practical exerc. exercises

The overview about the whole semester was missed.

Postscript ~~#4~~

Suggestions for improvements?

Designing interfaces (G. UI), exceptions signals.

- master solutions to all exercises

Answers to the exercises, explanations when we didn't get all points

Take a lecture more about types with more practical example, one that is revisited whenever a new aspect is revealed. Maybe drop Postscript?
Make Hudak paper homework and discuss after Fixed Points.

Would benefit immensely from repetition session with assistants to discuss the exercises!
Include references to modern languages!

It would be great to have such a course earlier. A bachelor's course about ~~the~~ the most useful programming languages at the time would be very useful.

The lecture to objects and types is, when you hear all this for the first time too difficult. Without Simons I wouldn't have the possibility to understand it. Why not treating this subject in two weeks? Or give as homework the lecture of Simons.

Lecture 9 was way to theoretical and hard to understand

Include some newer languages, if possible.

Make more exercises, or make them harder.
The students should be challenged, not bored.

Talk to the ~~in~~ people responsible for exam dates and reconsider a "holiday" of just 2 weeks before the exams. It would be extremely helpful.

Maybe lessen the subject matter.

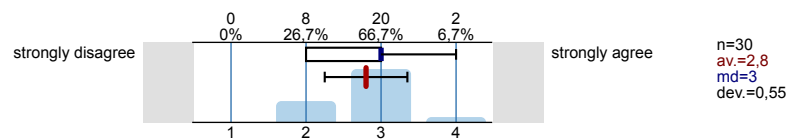
More explanations or notes on the slides, + more ~~extra~~ definitions, maybe an appendix of often used terms.

I'm not sure coding that much postscript is useful
Maybe see more of Python, Ruby etc.

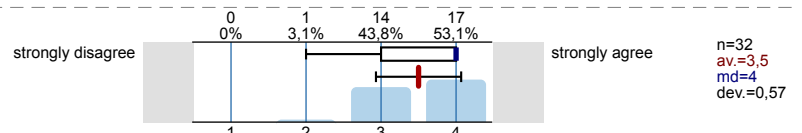
More functional programming - increasingly relevant in industry.

Assessment of Individual Lectures

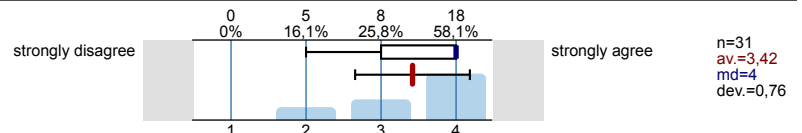
8.1 Introduction



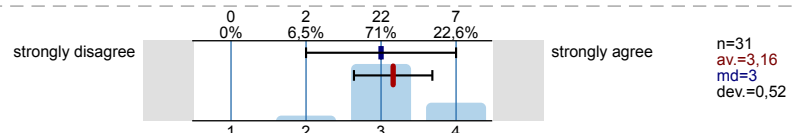
8.2 Stack-based Programming



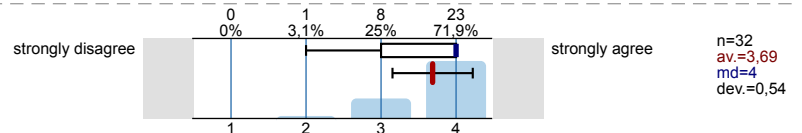
8.3 Functional Programming



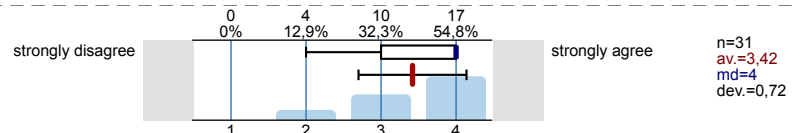
8.4 Types and Polymorphism



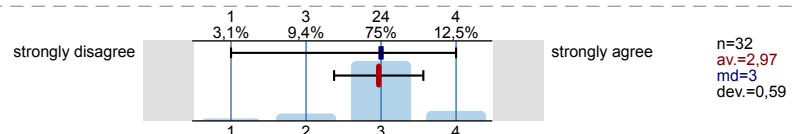
8.5 Lambda Calculus



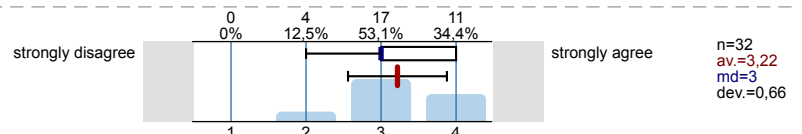
8.6 Fixed Points



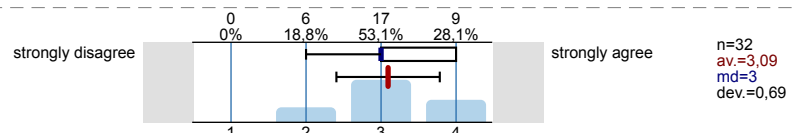
8.7 Programming Language Semantics



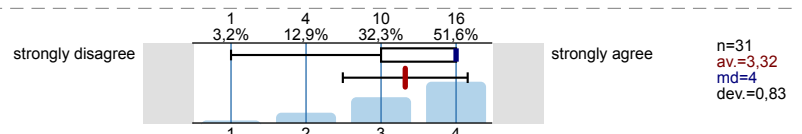
8.8 Objects and Prototypes



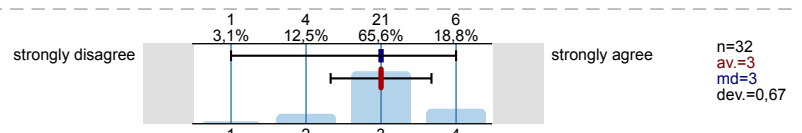
8.9 Objects and Types



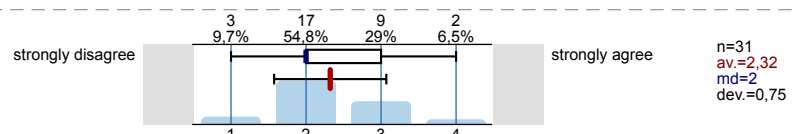
8.10 Logic Programming



8.11 Applications of Logic Programming



8.12 Visual Programming

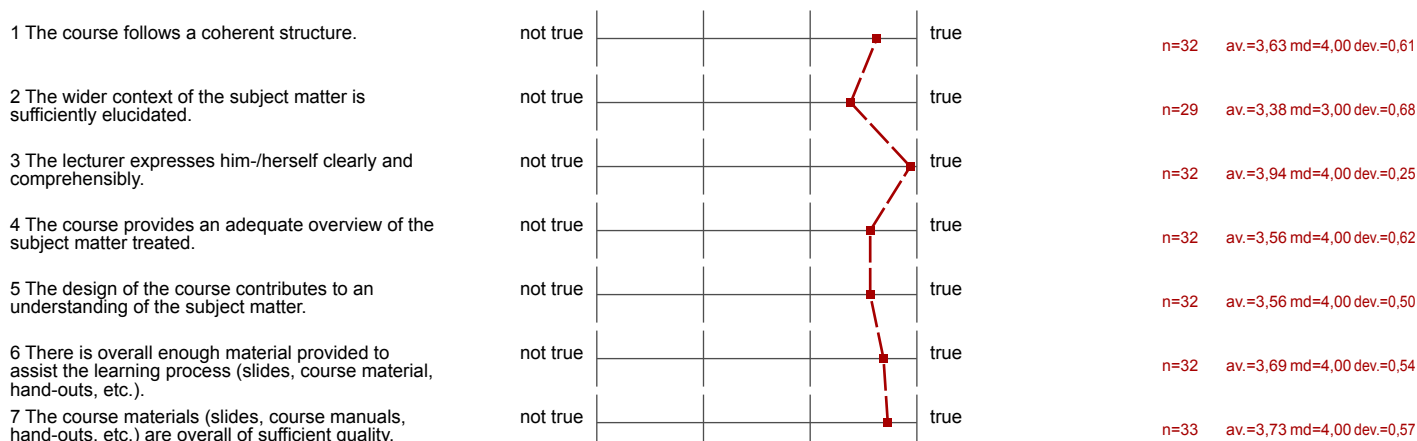


Profile

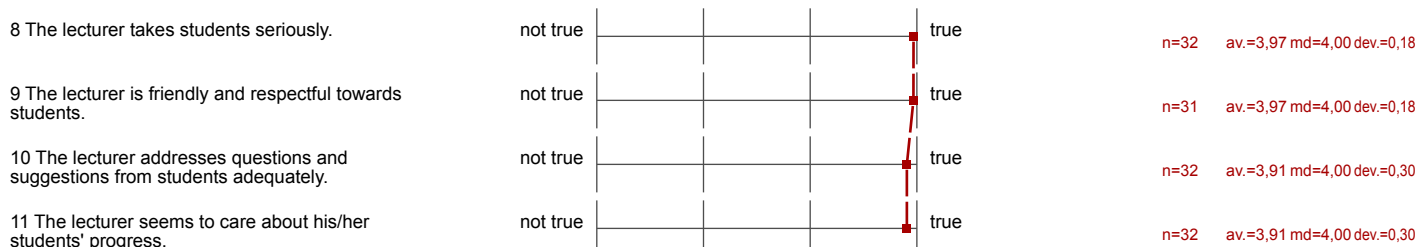
Subunit: Phil.-nat. Fakultät
 Name of the instructor: Prof. Dr. Oscar Marius Nierstrasz
 Name of the course: Programming Languages
 (Name of the survey)

Values used in the profile line: Mean

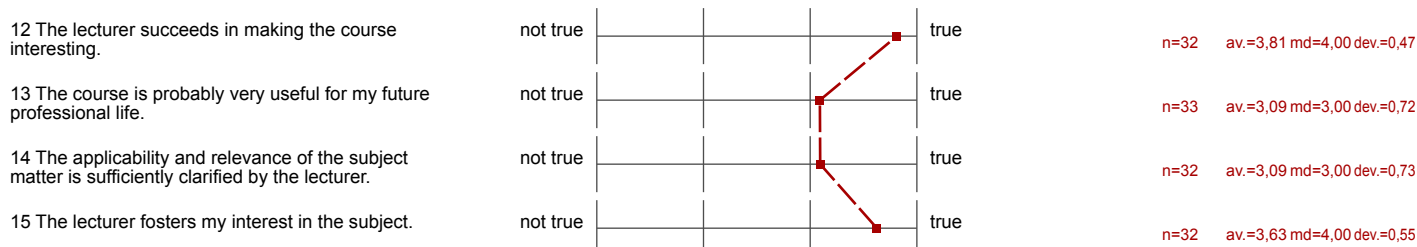
Planning and Presentation



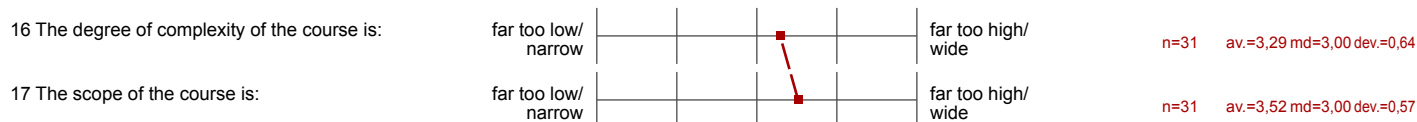
Manners with Students



Interest and Relevance



Complexity and Scope



18 The pace of the course is:		n=31	av.=3,35 md=3,00 dev.=0,66
19 The amount of knowledge presupposed by the course is:		n=31	av.=2,97 md=3,00 dev.=0,55

Overall Assessment

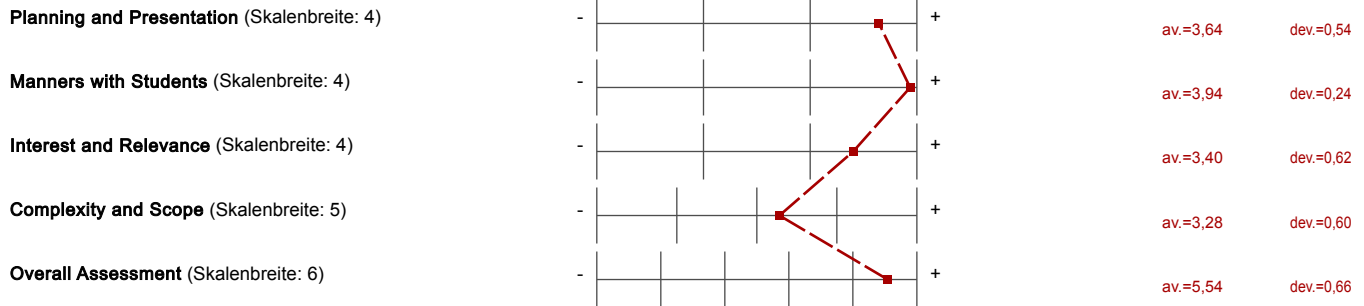
20 How would you grade the course as a whole?		n=30	av.=5,20 md=5,00 dev.=0,66
21 How would you grade the lecturer with regard to subject expertise?		n=30	av.=5,80 md=6,00 dev.=0,76
22 How would you grade the lecturer with regard to teaching methods?		n=30	av.=5,63 md=6,00 dev.=0,56

Assessment of Individual Lectures

8.1 Introduction		n=30	av.=2,80 md=3,00 dev.=0,55
8.2 Stack-based Programming		n=32	av.=3,50 md=4,00 dev.=0,57
8.3 Functional Programming		n=31	av.=3,42 md=4,00 dev.=0,76
8.4 Types and Polymorphism		n=31	av.=3,16 md=3,00 dev.=0,52
8.5 Lambda Calculus		n=32	av.=3,69 md=4,00 dev.=0,54
8.6 Fixed Points		n=31	av.=3,42 md=4,00 dev.=0,72
8.7 Programming Language Semantics		n=32	av.=2,97 md=3,00 dev.=0,59
8.8 Objects and Prototypes		n=32	av.=3,22 md=3,00 dev.=0,66
8.9 Objects and Types		n=32	av.=3,09 md=3,00 dev.=0,69
8.10 Logic Programming		n=31	av.=3,32 md=4,00 dev.=0,83
8.11 Applications of Logic Programming		n=32	av.=3,00 md=3,00 dev.=0,67
8.12 Visual Programming		n=31	av.=2,32 md=2,00 dev.=0,75

Profile

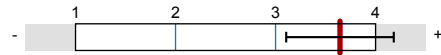
Subunit: Phil.-nat. Fakultät
Name of the instructor: Prof. Dr. Oscar Marius Nierstrasz
Name of the course: Programming Languages
(Name of the survey)



Presentation template

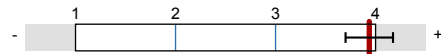
Programming Languages
Prof. Dr. Oscar Marius Nierstrasz
No. of responses = 33

Planning and Presentation (Skalenbreite: 4)



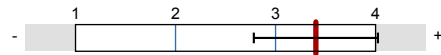
av.=3,64

Manners with Students (Skalenbreite: 4)



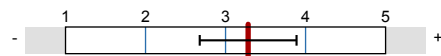
av.=3,94

Interest and Relevance (Skalenbreite: 4)



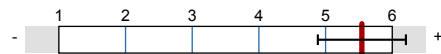
av.=3,4

Complexity and Scope (Skalenbreite: 5)



av.=3,28

Overall Assessment (Skalenbreite: 6)



av.=5,54