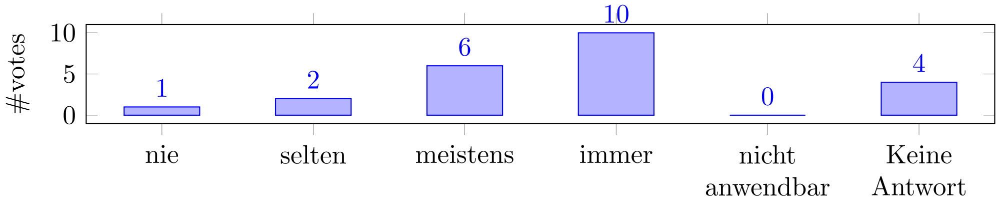
Modul: Introduction to High Performance Computing	**Semester:** WS 23/24

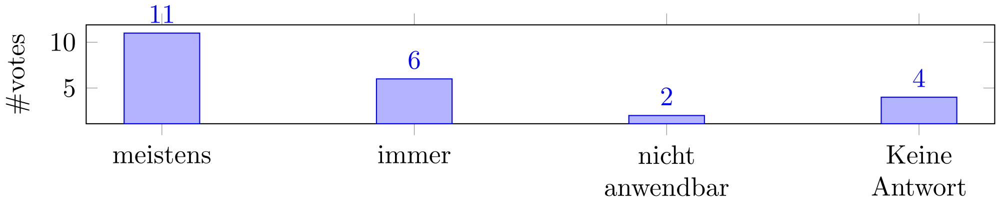
Ergebnis der Online-VLU. Die Umfrage fand in den letzten beiden Vorlesungswochen statt.

1 Bewertung der Vorlesung

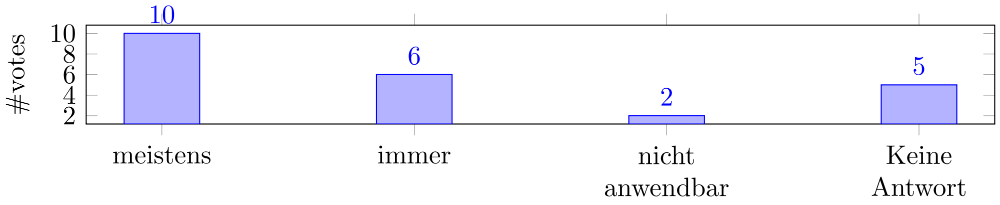
Wie oft hast du die Vorlesung besucht?



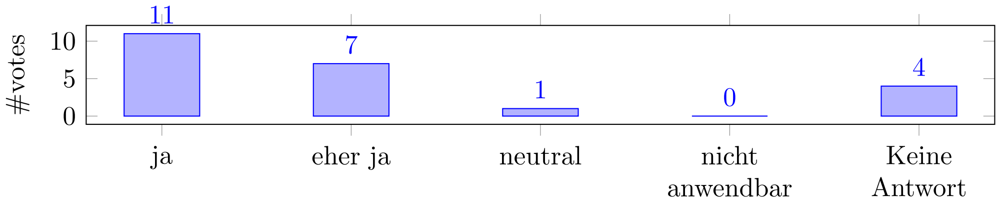
Wurden Themen durch Beispiele veranschaulicht?



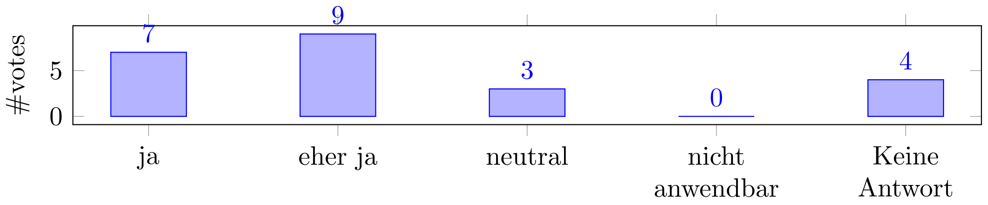
Wurden die Themen ausführlich genug erklärt?



War die Struktur der Vorlesung klar zu erkennen?

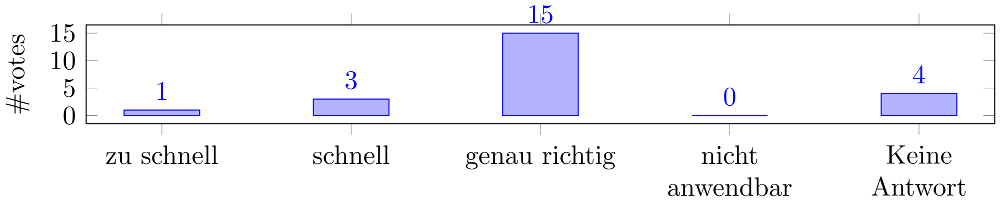


Waren die Folien/das Skript hilfreich?

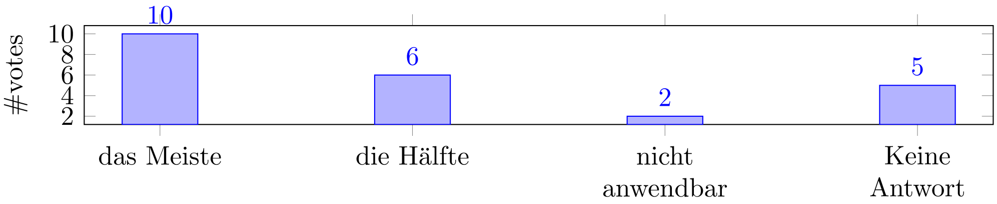


2 Bewertung der Dozierenden

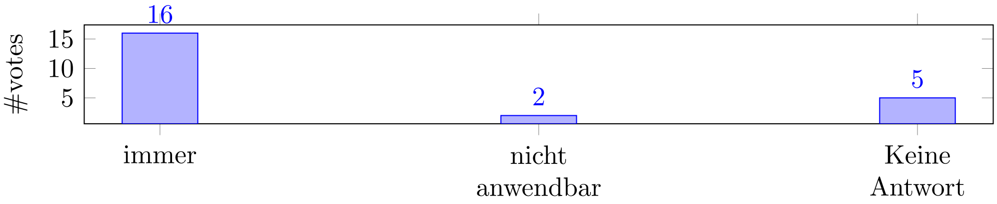
Die Geschwindigkeit der Vorlesung war...



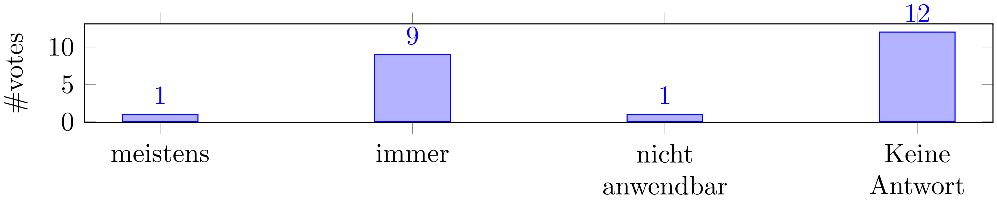
Wie viel verstehst du während der Vorlesung?



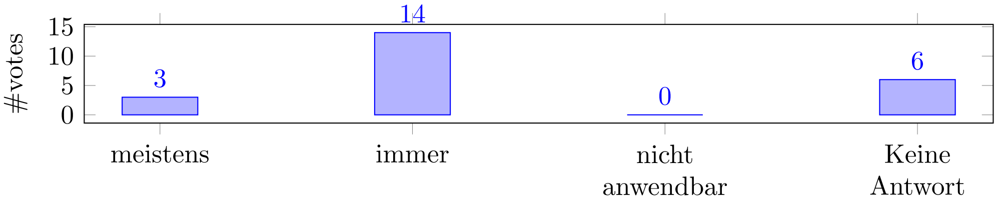
Ist der Dozent/die Dozentin gut auf Fragen eingegangen?



War der Dozent/die Dozentin außerhalb der Vorlesung für Fragen etc. erreichbar?

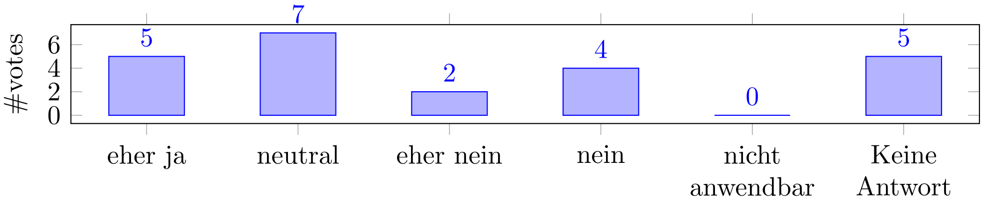


War die Dozentin / der Dozent akustisch gut zu verstehen?

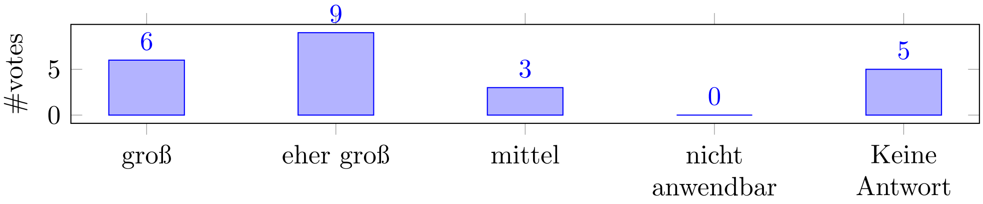


3 Bewertung des Moduls

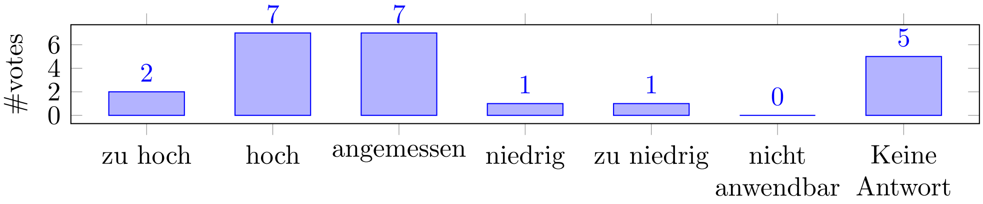
Findest du die verlangten Studienleistungen für dieses Modul angemessen?



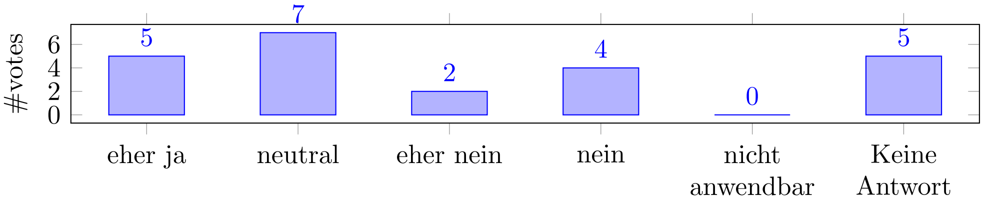
Der Praxisbezug war...



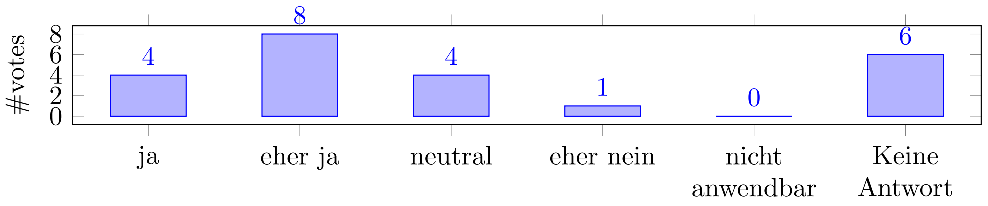
Ist der Arbeitsaufwand für dieses Modul im Hinblick auf die LP-Zahl angemessen?



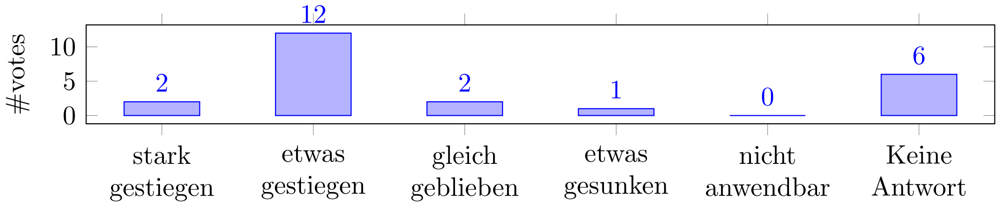
Findest du die verlangten Studienleistungen für dieses Modul angemessen?



Würdest du dieses Modul weiterempfehlen?

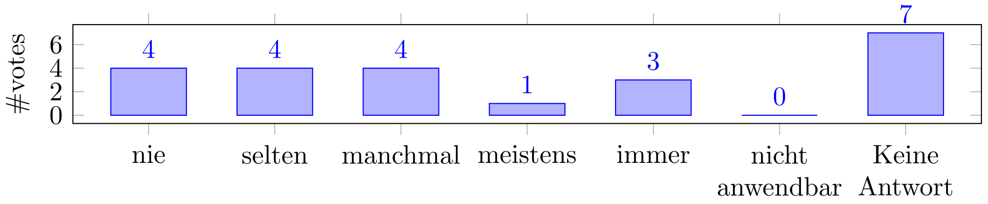


Dein Interesse für dieses Thema ist...

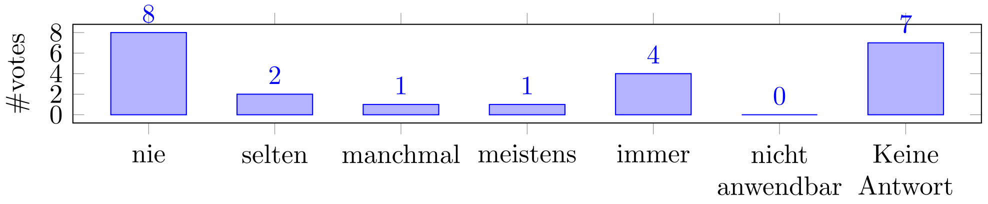


4 Bewertung der Übungsaufgaben

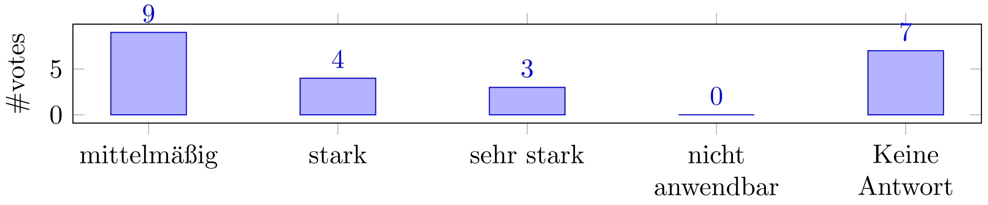
Wie oft hast du die Übungen besucht?



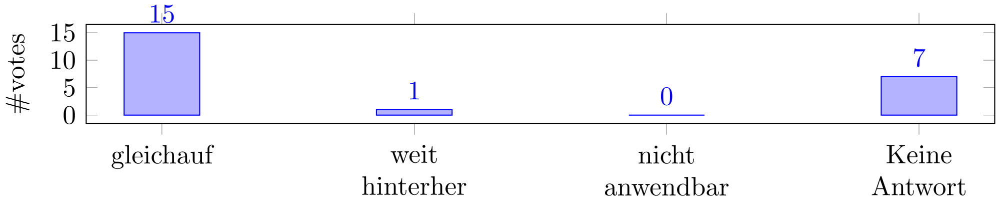
Wurden die Übungsaufgaben rechtzeitig zur Verfügung gestellt?



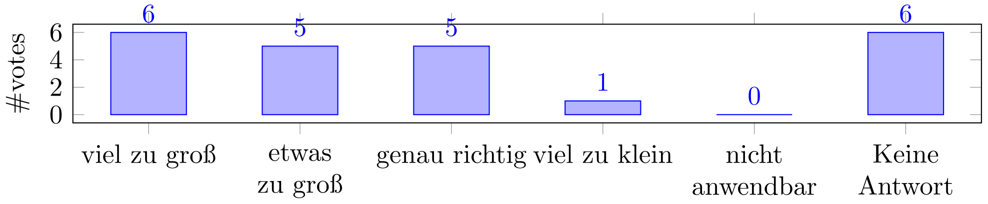
Die Schwierigkeit der Übungsblätter schwankte...



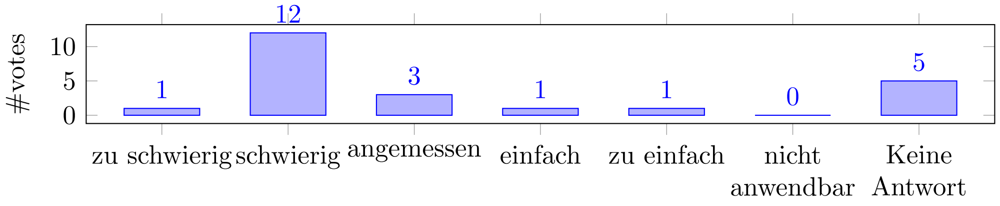
Die Vorlesung war...



Die Übungsgruppe war...

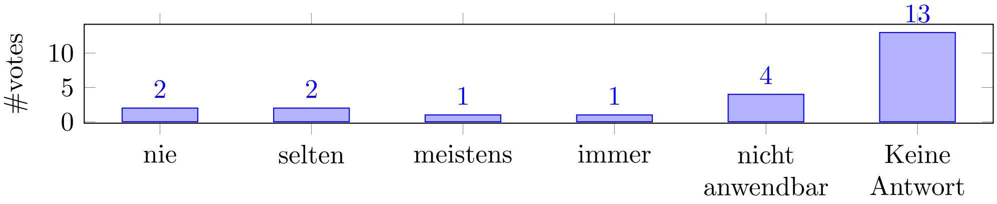


Die Übungsaufgaben waren meistens...

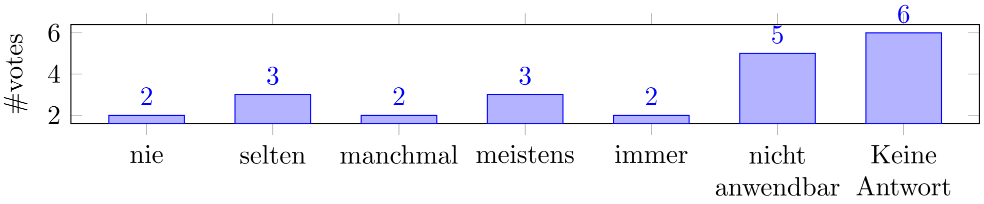


5 Bewertung des Tutoriums

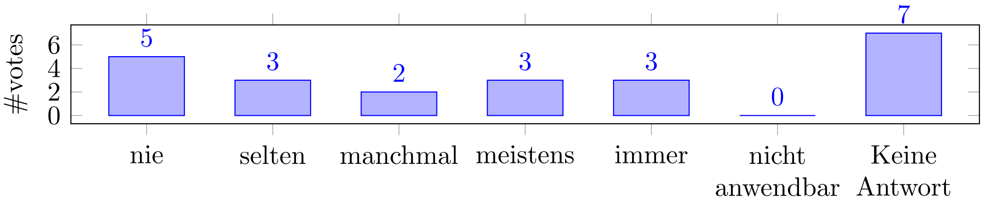
War der Tutor/die Tutorin außerhalb der Übung für Fragen etc. erreichbar?



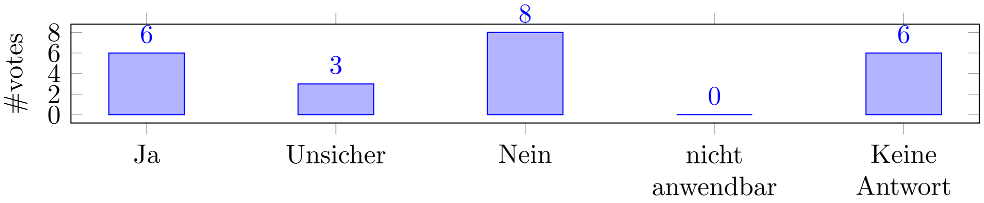
Waren die Korrekturen des Tutors/der Tutorin nachvollziehbar?



Wurde der Tutor/die Tutorin mit dem Stoff der Übung fertig?

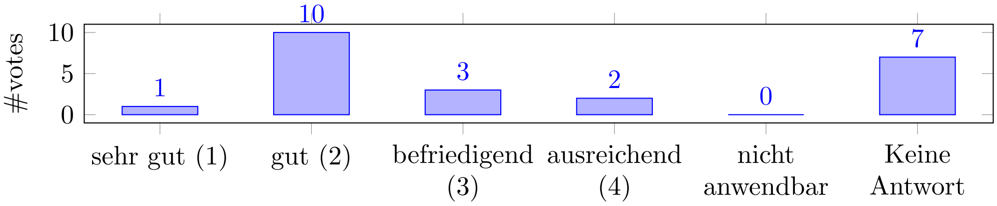


Lohnt sich der Besuch des Tutoriums?

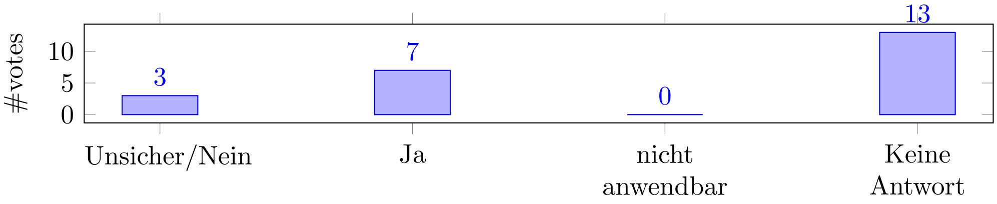


6 Abschließende Bewertung des Moduls

Note:



Hältst du die Vorlesung der Dozent:in für lehrpreiswürdig?



6.1 Wieso?

Good, easily understandable lecture despite a relatively dry topic. Had fun with the audience. Was knowledgeable about the topic.

The atmosphere in the lecture hall was always very good. The slides were well structured. The topics were interesting. There was always time for questions. I could go on, the lecturer and the lectures were really good.

hell yeah, lecturer is always cheerful which reflects on us too!

Her lecture was most of the time interesting (except 2 lectures out of ˜12 because i felt that we're learning the API which simply can be looked up :P

the slides were very good & she does a small break in between which helps after hammering lots of information into us.

7 Freitextkommentare

7.1 Was hat dir an dieser Lehrveranstaltung gefallen?

very interesting topic, well presented and taught in a very practical and hands-on way

Lecture

Practical applications in the exercises. Content and presentation.

Getting to actually run code on the HPC machines.

The atmosphere in the lecture hall was always very good. The slides were well structured. The topics were interesting. There was always time for questions. I could go on, the lecturer and the lectures were really good.

the lecture was fun and had good amount of knowledge without it being too overbearing.
It was well paced and in bite sized chunks.

The lecture was really good and I enjoyed the explanations and they way of teaching of the lecturer

Actually using the super computer for solving tasks.

lecturer & content

- presentation style

- very interesting topic
- nice presentation style
- if there were any open questions they were answered in detail in the next lecture

Lectures and corresponding slides are well-organized with deep details.

Die Vorlesungsfolien fand ich gut und die Inhalte spannend

7.2 Was könnte noch besser gemacht werden?

I think the exercise sheets contained quite a number of errors. Usually only a couple, but that undermines the trust in the given (multiple-choice) solutions. Which makes it difficult to determine whether I should re-do the working out and spot my error. Or if the solution is off again and I just need to guess which one is considered correct.

The tasks in the tutorial: The tasks should be more about how to program and less about how to plot the results.

Style of exercise tasks (Less programming of diagrams. Focus more on understanding and explaining things would be more useful)

Split exercise in smaller groups. More balanced workload.

Exercise sheets were too long because repetitive tasks.
I learned more about plotting data with matplotlib than the actual tools I was supposed to use, at times.

The effort for the exercises was extraorbitantly miscalculated. It took 20-30 hours for some of the exercises. In my opinion, this is not justifiable for a course. Although the exercises had a very strong practical relevance, they prepared you very little for the exam, which makes the time required even more ridiulous.

That's why I have very mixed feelings about recommending the *course*, because I actually really enjoyed the *lectures*.

The excerise difficulty for the time allocated (1 week) to finish it sometimes were a bit too much.
I would also like to have the results from the excersice sheet available for comparisons after we see them in the tutorial, not the full solution but maybe the output graphs to be able to check for myself at later time and if i had questions about them, because looking at the solution in the tutorial isnt enough for me to memorize it and compare it (yes i can take a picture of it but i would prefer to have it available) also for the problem solved that requires formula to be able to know how did you use the formla in practice because sometimes there is no example in the lecture slides.

For sure the exercises as well as the admission.
I do not think its helpful to do plots over and over again, since I did not learn much from it.
Additionally I do not think the presentation admission makes sense when having only one tutorium with a lot of students, which mostly are just there to shortly presented their results boringly to get the admission.
I would rather have an exercise session where a real exchange can happen.

Exercises should be redone completely:
- the ”tutors”had no idea about the topic (even with easy stuff like OpenMP/MPI/Cuda Hello Worlds) and were just orchestrating who would present
=¿ even if something wrong was pointed out they did not show any kind of understanding
=¿ maybe they should actually do the tasks themself instead of reading of the solutions

- the exercise sheets were very annoying with the number of intermediate steps
=¿ instead of splitting up tasks, maybe add a bigger more complex task with more freedom
- furthermore some tasks were very repetitive (like typing lscpu three times and answering the same 8 questions)

- at last the organization was awful
=¿ we did not have a full week for a sheet (Friday to Wednesday) + if the server is unavailable (ecampus) it's not an excuse to deny solutions!
=¿ Filling in the ecampus check list took too much work (up to 40 questions each time; was better at the end)
=¿ the auto grading did not work at all for the fist few sheets (was better at the end)
==¿ I think it would actually take less time to correct assignments by hand if students formed groups of 2/3/4

---

wasn't really happy to do the exercises but at the same time it's difficult to pick good exercises for this module. so i don't know

---

- Split up the exercise into more smaller groups
- Get rid of the ”checklistön ecampus
- Reduce the amount of (or improve) plotting exercises (basically all exercises were in the same manner and very repetitive: run this, plot this, interpret this; making the important interpretation task harder to get to, when you made a mistake)
- Deliver script files for the executables
- folder hierarchy / the way you hand in solutions (folder for every exercise and its subtasks instead of having painfully long filenames)

- split up tutorium group (probably didn't anticipate that many people)
- a large part of the later exercises was creating plots which was repetetive (maybe provide scripts and focus more on the interpretation)
- not a fan of the ecampus checklist (many questions like "did you do task xy/ at least programming tasks could be done with CI/CD only)

Absolutely tutorials... I understand the idea of "presentingßtudent's solution (I have no problem with this one). However, I think the correct solution and corresponding explanations always required by tutors (adding few comments to presenter is not enough). (i.e. If we see a graph, I want to learn all possible underlying reasons of the graph. I don't want to come to tutorials to hear "yeah, you see this is the graph. That's it bye."). I think tutorial was better close to the end of the semester (at least there were more comments on why we those results etc.).

Die Übungsaufgaben waren häufig nur tooling schlacht und der Lernerfolg war, wie man am besten langweilige Aufgaben wegautomatisiert. Die späteren Übungsaufgaben haben mehr Boilerplate Code bereitegestellt, die das Problem verbessert haben. Bei der nächsten Vorlesung sollte daher mehr Hilfe bei der Ausführung der Aufgaben gegeben werden, damit sich auf den Kern der Aufgabe fokusiert werden kann.

The exercises were just a joke. They took like 6 hours a week, but didn't deepen the understanding of the lecture content accordingly. Often there were questions like look into documention x and read some values there, which just felt a bit pointless. But the real problem were the programming exercises, which took an exessive amount of time (Looking for bugs, compiling and plotting). And after hours of just frustrating work, one had to fill out questions on an eCampus checklist, where a lot of times the given answer options were all wrong. I understand the intention of doing practical exercises, but the execution was just bad. To the end, where the new tutor came, the exercise sessions were not that pointless as before because he explained a lot of the exercises and the intention behind them. For next year it might be better to do less programming exercises but, to give some plots or programs and to ask the type of questions the new tutor asked in the exercise session (what do you see in the plot and WHY do you think xy looks like it does? how would it look if we would change xy?)

7.3 Hier hast du Platz für weitere Anmerkungen und Feedback zum Modul.

The trip to marvin was great :)
I think I would have liked a final, more system-level programming task: simd + openmp + mpi. Maybe build it up over the series of exercises and combine them in the end

Might be better to expand content a bit and make it a 9 credit module

One last thing mentioning is the way how people are chosen to presented their solution in the exercise.
I think (even if I was not affected) its very unfair to priotize people sending a mail with the wish to present beforehand without communicating this possibilty beforehand.
Since other people went to the exercise session multiple times trying to present and still needed to be there 5 or more times without enjoying the exercise session while other simply can write a mail and tell Oh I want to present next time.
Please find a way to make this more fair for all students!

- module couldve been way better if not for the exercises

First, let's quickly talk about the good things. The lectures. The lecturer is awesome, very good at presenting their beloved topics, clearly structured, the slides are perfectly fine (and mistakes get corrected upon recognizing them / immediately), the lecture material is always uploaded on time and one can ask stuff in and outside the lecture. The lecturer even does a lot additional research at home, if needed and presents it at the beginning of the next lecture, if they cannot answer the question right away. That's what I call commitment. Well done!

The bad: The Exercises. Exercises were done for the first time for this module. Thus for all the feedback one has to take into account that the first iteration of exercises is always rough to master.

Exercise sheets were often very time consuming and frustrating. They contained, especially regarding the earlier ones, a lot of mistakes. The solutions had to be given on github for code and on ecampus for answering questions (which were often correlated with the practical coding sessions). Often, the set of possible solutions didn't include an actual valid solution, making the questions rather a guessing game than anything worth one's time. Besides that, many exericses across the first 9 sheets were only 'fill in code lines in predefined unfinished programs'-type of exercises for which you need to find the correct functions of various libraries. Many other tasks involved just executing commands without any need of understanding what is happening or simply plotting a lot of different graphs. Which both might be time consuming as well as questionable in regards of actual relevance for learning the cool HPC stuff.

I think the exercises need to be reviewed a lot for a second iteration, definitely for revising mistakes regarding inprecisely formulated tasks, wrong information given in the tasks and the incorrect sets of answers for the checklists on ecampus.
Maybe it could be considered to change the format of the exercises completely. Regarding the fact that this course is pretty popular now that it is a 6CP module, one can also clearly see that the amount of participants was a bit overwhelming for the tutors. I don't know a solution for this except for maybe just publishing less exercise sheets in total but make each one of them focus on a topic from the bottom up. One exercise sheet did this quite nicely where we had to program one concept almost entirely by oneself and answer somewhat useful questions about it, instead of just executing a set of command lines, and plotting their data a thousand times. I just think that coding from scatch with some clear instructions is a better as a learning tool for the practical parts of HPC.

Also, I think that the theoretical part of the exercises was a bit too low. This results in the exercises preparing you just faintly for the examination. In my opinion, the exercises should aim at preparing you for the exam, strengthening your understanding for what is about to be questioned in the exam. Since the exam is held in the timeslot of the very last lecture, one has little time to learn the entirety of theoretical HPC stuff, if code and practical stuff is explicitly not asked during the exam.
Maybe a 50/50 ratio of practical and theoretical exercise tasks would be beneficial for the students, maybe less but slightly bigger sheets (in comparison to the size of sheets of this semester) would be beneficial for the tutors (as they deliver more time flexibility) and the students (because of the same reason). But surely I don't know for you have more experience in organizing a module than I do.

Also, some other notes regarding the exam admission:
1. There were some frustrating elements to that, but I am sure that everybody knows about them already. I am sure that the exam submission for future iterations will be dynamically well planned from the beginning of the course, once the amount of interested participants is roughly clear. So I will not go into any detail here.
2. Although it did not affect me personally, I think that the concept of writing emails to be prioritised for presenting exercise tasks (which seemed to be a necessary requirement for the exam admission) is simply unfair and should be reconsidered for the future. In every single exercise session, people challenged their luck to be chosen by the tutors for the opportunity of getting their exam admissions by presenting solved / partly solved tasks. Some people were so unlucky that they sat there for several fridays just to not be chosen a single time. That's all fair so far, nothing that can be improved unless the exam admission will be changed. But: When people firstly had the idea to beg per email to be prioritised in this choice of who is the lucky one to present, this is just unbelievably