



Aalto University
School of Science

CS-E4640 Course Management

Hong-Linh Truong

Department of Computer Science

linh.truong@aalto.fi, <https://rdsea.github.io>

Lectures, tutorials and meetups

- **Lectures**
 - Key concepts about principles, models, methods and technologies
- **Tutorials**
 - Practical, concrete tools and hands-on discussions
- **Meetups**
 - Adhoc, not mandatory contents but useful tips/experiences
- **Nr. of lectures + tutorials != Nr. of slots in the course agenda**
 - Backup dates (e.g., in case of sickness) & on-demand face-to-face discussions

All dates in the agenda must be booked!

Schedule

Remember the schedule:


<https://version.aalto.fi/gitlab/bigdataplatforms/cs-e4640/-/blob/master/schedule.md>

Wed: 16:15-18:00 (lecture)

Thu: 10:15-12:00 (hands-on/meetup)

We try to reduce the online lecture time and use remaining time in Wed/Thu for discussion

Communications

- **Course discussion (no moderation!)**
 - Microsoft Teams (pls. register, see the link in MyCourses)
- **Find the ideas/answers from the Internet – no problem**
 - Everyone knows  **stackoverflow**
- **Everyone should help sharing the knowledge w.r.t course topics**
- **We try to react as soon as possible but don't expect real-time!**

The Teaching Assistant team

Minh-Tri Nguyen

PhD student

tri.m.nguyen@aalto.fi,

<https://research.aalto.fi/en/persons/tri-nguyen>

Linh M. Nguyen

Master student

linh.m.nguyen@aalto.fi,

<https://www.linkedin.com/in/my-linh-ng/>

Reaching them via Teams!
(avoid a lot of emails)

Unfortunately, the staff is under-resourced this Spring due to the lack of TA candidates!

Personal discussion

Due to COVID – only online meeting:

- **Using Microsoft Teams to chat and get meeting slots for one-to-one calls**
- **Discuss your problems with the professor in charge**
- **Try to have personal discussion with our Teaching Assistants as well!**
- **Share your problems in the Teams so that we don't need to repeat the similar questions**

Assignments

- **3 assignments**
 - Each divided into 3 parts (design, implementation, and extension)
 - Within a part: an objective is evaluated in the 0-5 scale, then multiplied by a pre-defined weighted factor (based on the part)
 - **No final exam!**
- **Assignment evaluation**
 - Real world development, reporting, and demonstration
 - No automatic grading: we will check your code and do **reproducible test**

Assessment for each assignment

- **Software artefacts**
 - e.g., code and configuration
- **Data**
- **Written reports in Markdown** (<https://en.wikipedia.org/wiki/Markdown>)
 - *For explaining design, evaluation and installation*
- **Records of running results: logs/screenshots**
- **Each part might have a weighted factor of 2 or 3 (e.g., $5 \cdot 3 = 15$ points, with weighted factor=3)**
- **An assignment should be managed as a git project by yourself**

Assignments

- **Academic honesty**

- Follow the university rule, peer discussion is OK but creating your own solution
- Check the consequence of academic violations here
<https://version.aalto.fi/gitlab/bigdataplatforms/cs-e4640/-/blob/master/violations.md>

- **All deadlines are hard**

- **You might be requested to have a face-to-face to discuss your assignment results, e.g., when we are not sure**

- you understand your solution or how to reproduce the results of your solutions

Final grading mapping

Highest	Lowest	Letter
100.00 %	90.00 %	Excellent (5)
89.99 %	80.00 %	Very Good (4)
79.99 %	70.00 %	Good (3)
69.99 %	60.00 %	Satisfactory (2)
59.99 %	50.00 %	Pass (1)
49.99 %	0.00 %	Fail (0)

Flexibility versus limitation

- **Can use Java, JavaScript/TypeScript, Python and shell scripts only**
 - We are elastic but we cannot handle all possibilities
- **Use the recommended dataset and technologies**
 - But you can propose your own dataset
- **Deadlines are hard (don't be surprised!)**
 - We cannot be flexible in order to guarantee the grading on-time
 - Special exception handling is case-by-case (e.g., sickness, family issue)

Resources

- **Check hints from the course Git/Mycourses**
 - E.g., Git assignment templates/examples and references
- **Computing infrastructures and data**
 - Google Cloud Platform: everyone gets 50USD credit
 - Many tests can be run in your own computers with virtualization technologies enabled
 - Try to use Cloud free services
 - CSC if you can get the resource: <https://rahti.csc.fi/>

“I don’t take a computer science major! ”

- **Not all of you need everything**
 - Just want to learn analytics atop big data platforms?
 - *E.g., too much “systems” in Big Data Platforms!*

→ what would be the best strategy to learn this course?

Thanks!

Hong-Linh Truong
Department of Computer Science

rdsea.github.io