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 and tutorials

- **Lectures**
  - Key concepts about models, methods and technologies
- **Tutorials**
  - Practical, concrete tools and hands-on discussions
- **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! You will be informed about no lecture/tutorial day around a week/2 weeks in advance.

Communications

- **Course discussion (no moderation!)**
  - Slack channel
  - Online forum discussion in MyCourses
- **Find the ideas/answers in the Internet – no problem**
  - Everyone knows stackoverflow
- **Everyone should help sharing the knowledge w.r.t course topics.**
- **Don’t expect that I will reply a request in real-time!**
  - *Wed and Thu are my most active online days for the course’s related topics*

Personal discussion

My office

- **Room A231, CS Building (Konemiehentie 2, 02150 Espoo, Finland)**
- **For course related topics: try to catch me on Wed/Thu**

Assignments

- **Four assignments**
  - Each with 25 points, divided into 2-3 parts
  - 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
  - Almost no automatic grading but we will check your code and do reproducible test

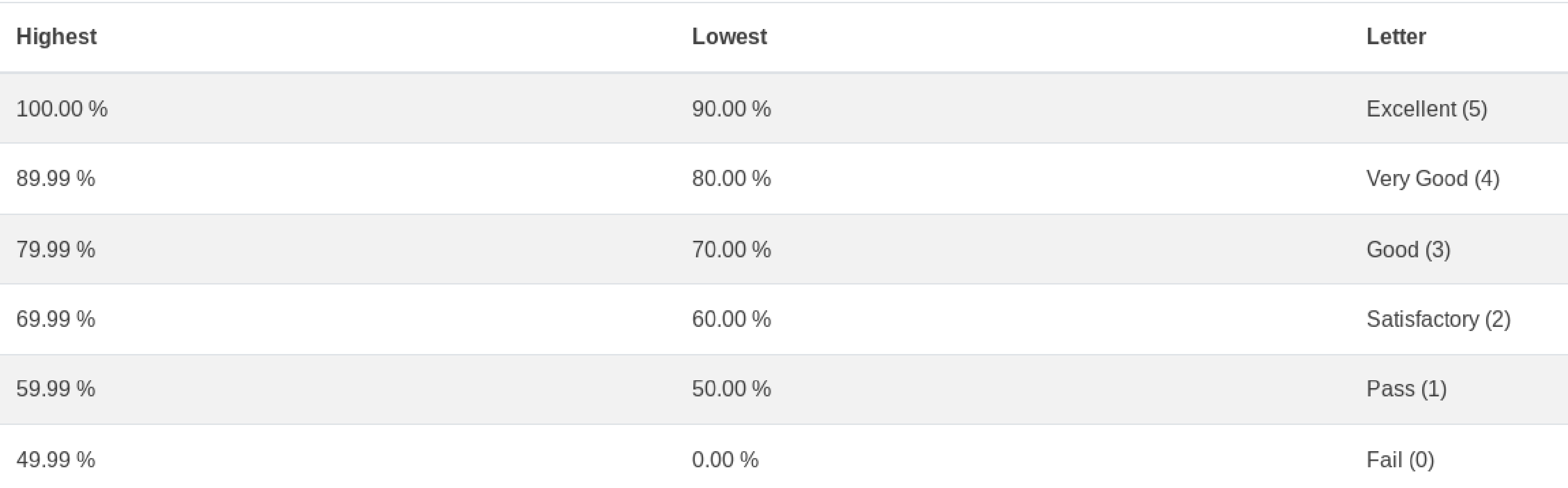
Assessment for each assignment

- **Total: 25 points**
  - Software artefacts: e.g., code and configuration
  - Data
  - Written reports in Markdown (https://en.wikipedia.org/wiki/Markdown)
    - *For explaining design, evaluation, installation*
  - Records of running results: logs/screenshots
  - Each part might have a weighted factor of 2 or 3 (e.g., 5*3 = 15 points, with weighted factor=3)
- **Assignment should be managed as a git project by yourself**

Assignments

- **Academic honesty**
  - Follow the university rule, discussion is OK but creating your own solution
- **All deadlines are hard**
- **We will leverage “big data platforms” concepts to manage your submitted outputs (code, logs, etc.)**
- **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
  - 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, Scala, JavaScript/NodeJS, 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**
  - 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 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
  - Aalto machines
  - Try to use Cloud free services
  - CSC if you can get the resource: https://rahti.csc.fi/

Thanks!

Hong-Linh Truong
Department of Computer Science

rdsea.github.io