TECHNISCHE UNIVERSITÄT MÜNCHEN
SCHOOL of COMPUTATION, INFORMATION and TECHNOLOGY

Language Desing: Internals of Kotlin

Kickoff

Dr. Michael Petter, Yannick Stade

SS 2023

Section 1

Organization

Schedule

Jan. 31st	Pre-course meeting
Apr. 3rd	Kick-off meeting with topic revealing
Apr. 11th	Submission of topic ratings
Today	Introductory meeting
Until May 12th	At least 1 individual meeting (contact your supervisor to set a date, send any references you want to discuss in advance)
Jun. 14th	Draft paper submission
Jun. 19th	Review submission
Jun. 28th	Final paper submission
Jul. 12th	Talks

Deliverables and Grading

- Report (40%)
  - 5-7 pages
  - use the provided double-column LaTeX-template
- Presentation (40%)
  - 20-25 min
- 2 Reviews (20%)

- We will use grading rubrics (and let you know beforehand)
- It is mandatory to be present during the 2-3 days of presentations

Section 2

Literature

Citable Literature

Good to use

- Papers (conf./journal)
- Books, book chapters
- Published articles
- Manuals

Try to avoid

- Websites, Blog articles
- Wikipedia
- Advertisements
- Lecture slides and notes
- Source code

Finding literature

- Kotlin KEEP
- Starting points: DBLP, IEEExplore, ACM DL, Google Scholar, ...
  - Select appropriate keywords
  - Many papers/books accessible freely via the library / TUM VPN/Proxy
- Graph algorithms
  - Publications of the same author(s)
  - Publications at the same venue
  - Cites ... (listed references)
  - Cited by ...
- Relevant conferences: POPL, PLDI, ICFP, OOPSLA, (TACAS, CAV)
- Another starting point: your advisor

How to read a paper

Run 1:

- Abstract
- What does the paper present? (technique/tool/...)

Run 2:

- Abstract + Introduction + Conclusion
- Skim the rest, no details

Run 3:

- Full text in detail

How to read a paper

- Keep notes and questions as you read
- While reading, try out concepts/codes, recompute/check examples
  - annotate/highlight the paper, on whatever suits you

  → do not blindly accept everything for granted
- Try to summarise it with your own words
  - don't copy or look at the abstract
- Make a list of contributions & limitations
- What are the key ideas and insights?
  - may not be the same!
- What is new?
  - you don't have the background, but try to 'derive' from the paper/related work itself

Section 3

Scientific Writing

Writing Style

- Writing for Computer Science, *Justin Zobel*, Springer 2014
  ($\rightarrow$ online access via `https://opac.ub.tum.de`)
- Factual, precise, focused
  - Stay on topic, no storytelling, ...
  - Limit to important and necessary topics
  - Don't omit necessary prerequisites

$\rightarrow$ hold back on baseless opinions and presumptuous phrasing (*"it is well established, that"*)

- Avoid forward references
- Avoid *I*, prefer *we* (or passive voice)
- 'We' only describes the authors, not the reader

Citing

- All work that is not yours must be cited
  - Clearly describe the source
  - But: no wrong/inaccurate attributions
- Citing styles:
  - Literal (direct) quote
  - indirect quote (rephrase) $\leftarrow$ strongly preferred
- Exception: foundations can be assumed (generally first few Bachelor semesters)

Citing: Examples

The x86 architecture defines the register CR2 [1].

```
The x86 architecture defines the register CR2
~\cite{intel2019man}.
```

The x86 architecture defines the register CR2. It can be used with the instruction MOV. [1]

```
The x86 architecture defines the register
CR2.  It can be used with the instruction
MOV.~\cite{intel2019man}
(paragraph)
```

Valgrind [1] is a tool for run-time instrumentation.

```
Valgrind~\cite{nethercote2007} is a tool for
run-time instrumentation.
```

Other approaches [1,2,3] ...

```
Other approaches~\cite{foo,bar,baz} \dots
```

Seminar Report

- like a *Review Paper*

- Abstract: brief summary of the area, problem, approach
- Introduction: problem statement, motivation, ...
- Background: required prerequisites
- Main part: summarize/explain different approaches, show applications/examples, evaluation, comparison, discussion
- Summary and outlook

Section 4

Reviews

Review

- short summary 1-2 paragraphs
- obligatory: positive feedback
- if necessary: ~~negative~~ critical feedback
  - in a **constructive form**,
  - if feasible with suggestions for improvement

→ do not shy away from critical feedback, as long as you stay factual

Section 5

Presentation

Content Selection

Presentation for the audience!

- What do you want the audience to take away?
  (Not: what can I talk about!)
- What are the key points?
- How much content fits into the time slot?

! Do not be afraid to reduce the amount of content, however:

- make sure that you mention at least some crucial contribution
- do not conceal problematic content
- introduce background information *by need*, not for the sake of it

Structure

For example:

- Motivation
  - Why is the topic relevant? Consider an eye-opening example
- Background
  - Consider referencing information from previous talks
- Concept
  - Use good/helpful examples, preferably *running examples*
- Evaluation
  - How good is the described concept?
  - Critical discussion of the topic
- Conclusions and outlook

Media

- Slides
  - For illustration purposes during the talk
  - Good to prepare elaborate examples and diagrams
  - Backup slides as preparation for questions
- Whiteboard, blackboard
  - Sticky place for permanently needed information
  - Helps to retrace the development of an example/diagram/code/algorithm execution
  - Answering questions
  - Spontaneously involve the audience
- Hardware, demonstration projects, etc.
- Check possibilities in advance

Slides: Style

- Title page: Title, name, institution, date, location
- On every other slide: number and title
- One topic per slide
- Avoid text
  - $\leq$ 10 lines
- Prefer graphics/illustrations
  - You may copy figures from the paper
- No unused points
  - Cover everything on the slides in your talk (i.e. not only mention but explain)
  - If not covered, remove!

Slides: Colors

- Few colors
  - Use colors sparingly, but systematically
- Sufficient contrast
  - Dark on white
  - Be careful with gradients
- Use special effects **only** when necessary
  - No annoying backgrounds (wave textures, etc.)
  - Animations only with sufficiently added value

Before the Talk

- Prepare slides, practice live examples/demonstrations, etc.
- Do a timed dry-run
  - Always recommended
  - Helps with uncertainity and time estimation
- Prepare on-site
  - Laptop, Beamer, laser pointer, clock, etc.

Talking Style

- Speak freely, with elevated volume
- Don't go too fast/slow
- Stay in contact with the audience
  - Eye contact, position, body posture, gestures etc.
  - Do not drink in the middle of a sentence, wait e.g. until a slide changes
- Usually at least 1 minute per slide
- Stay in time limit
  - Optional slides can fill time
  - Regularly consult a watch
- Stay calm, without radiating boredom or disdain