



Specialization Seminar

Cezary Kaliszyk

Outline

- **Outline**

- Topics

- **Step 3: Presentation**

In a nutshell

- Step 1: Choose a topic (today)
- Step 2: Research and experiment
- Step 3: Prepare the presentation
- ... Discuss with supervisor **in advance** and try before!
- Step 4: 25 minute presentation (usually including demo!) + 5 minutes of questions
- ... June 30: Present and ask questions to others (bonus points)!
- Step 5: Prepare a final report. Send one by July 15 to me (cezary.kaliszyk@uibk..)
- Step 6: Grade!
 - Combination of your slides, presentation, answers, report, participation.

Outline

- Outline

- **Topics**

- Step 3: Presentation

Step 1: Topics

- Lambda-Prolog
- J
- Kotlin
- Epigram
- Visual Basic
- Go
- Emacs Lisp
- Lua
- Perl
- Smalltalk
- Postscript
- Swift
- ...

Step 2: Research

Analyze

- limitation of languages
- distinguished features of languages
- the power of esoteric languages
- languages for dedicated purposes

Brief history and vision

Actual Use

Differences to common / known languages

- C, Haskell, Python, OpenGL, T_EX, Prolog

Outline

- Outline

- Topics

- **Step 3: Presentation**

Step 3: Prepare a presentation (1/2)

- Don't forget an introduction, related work, citations
- For other students (not for experts in your field)
- Try to make the presentations lively (DEMO!)
- Measure your time beforehand and observe it
- Think of questions for each talk
- Different students: Adapt your presentation accordingly
- You may think if it can fit a bachelor project already
- Is it also a good impression of you - as an interview?

Step 3: Prepare a presentation (2/2)

- Plan the presentation on paper before making the slides
- Tell us what this is about before introducing 10s of definitions
- Use illustrations / diagrams if possible
- Include examples to explain new concepts in their simple versions
- Do not make too long bullet lists (like this one)
- Number your slides
- Test your laptops before!
- Don't be late!

Step 5: Prepare a report (1/2)

Features

- describe the (non-standard) features of the languages in detail
- examples of use, simple introduction
- did these features make sense, where they used?
- did features of these language become standard in the meantime?
(functional programming was considered esoteric when it was first considered)
- how hard is it to program in this language?

Step 5: Prepare a report (2/2)

Dedicated Purposes

- necessity of creation
- what would programmers use, if the language would disappear?
- give (a set of) languages that could be used instead, explain your answer
- has the goal been achieved?
Logo was meant to teach kids how to program, has this had any effect?

Grade

Grade

- presentation (slides, actual presentation, answers, questions)
- seminar report

Presentations

- June 30, 25min+5min, in English

Report

- July 15, 10-15 pages, in English