Software Testing - Project

Prof. Dr. Sibylle Schupp[1], Sascha Lehmann[1]

[1]Institute for Software Systems/Institut für Softwaresysteme
Hamburg University of Technology (TUHH)

Summer Term 2022

Project Introduction

Welcome!

http://www.sts.tuhh.de

The Software Testing Project is brought to you by the

Institute for Software Systems

Lecturer

- Prof. Dr. Sibylle Schupp

Teaching assistant

- Sascha Lehmann

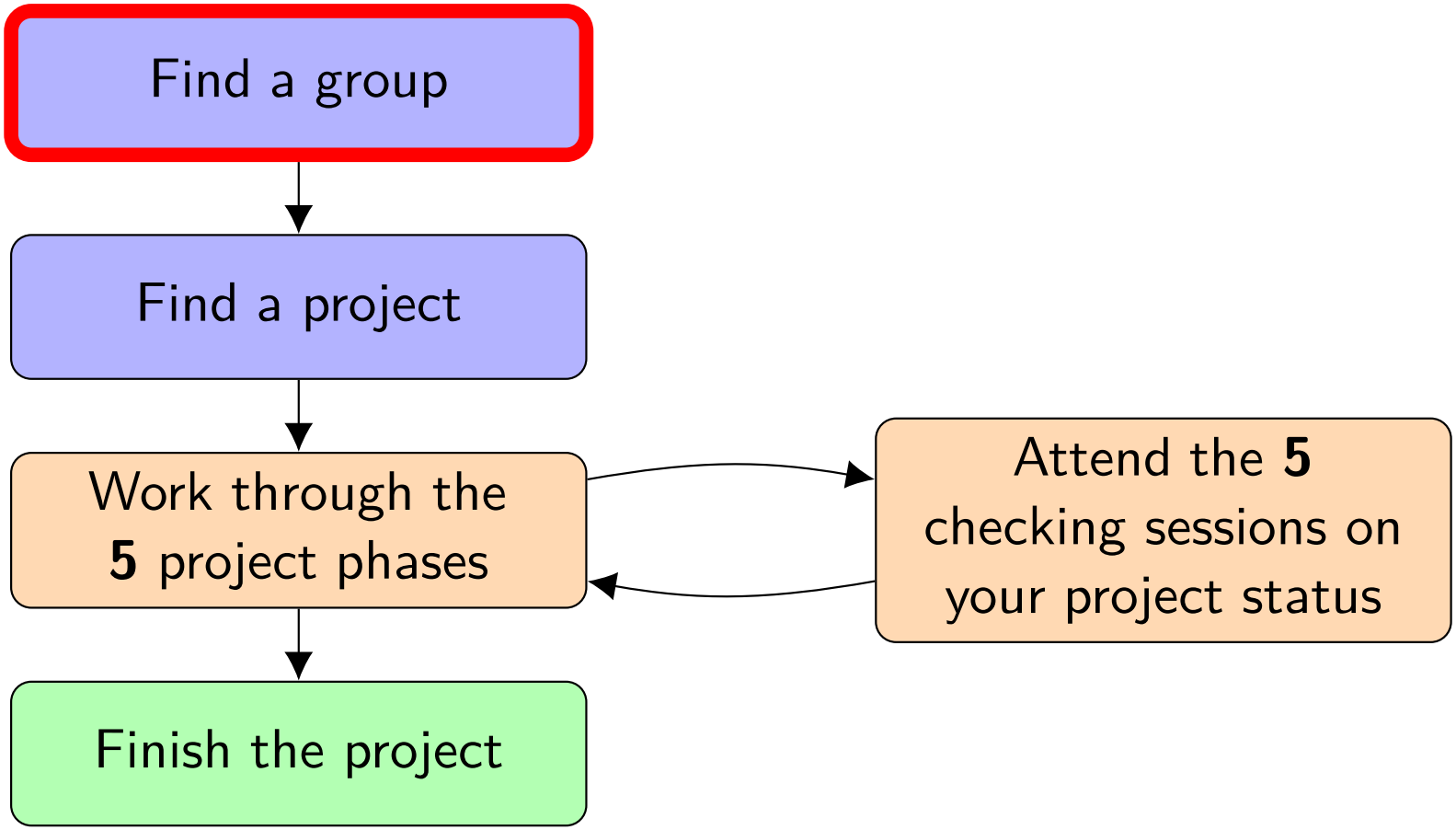
Organization

Project Prerequisites

Coding-intensive course at the **Master** level.

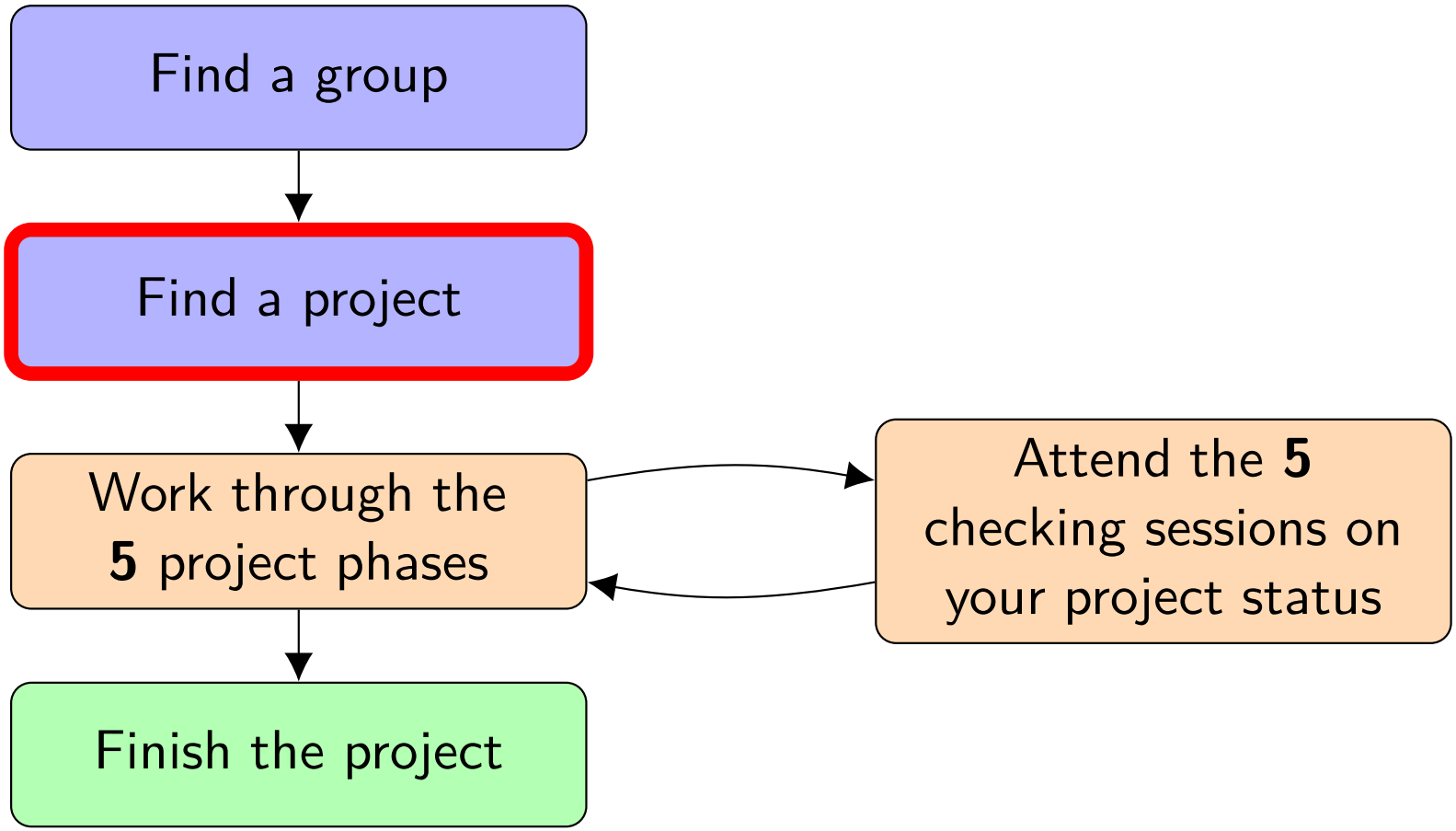
- Prerequisites in programming
  - Knowledge of object-oriented programming.
  - The subject system is in Java, the implementation language is Java.
- Prerequisites in software engineering/computer science
  - Project experience beyond homework assignments, e.g., through programming labs or small software projects
  - Knowledge of the fundamental data structures of computer science (stack, list, tables, tree, graphs)
  - Rudimentary knowledge of JUnit testing.
- All students with a IIW/CS Bachelor meet the prerequisites
  - Students with other degrees might have to catch up quickly
- Further, you must have the **time** to take the course

Project Structure



You can now sign up for one of the groups on **StudIP**!

Project Structure



Organization

How To Find Your Project?

- Once assigned to a group, you can **together decide** on your desired project (details on next slide)
- Check the group names on StudIP to see if the project was **already picked by a group**
- Send me (s.lehmann) an email with your selected project name and website link
- If the project was not chosen by another group before, and fulfills the requirements, it will be **assigned to your group**

Organization

How To Find Your Project?

Amongst others, you may find suitable projects via:

- Github[1] → e.g., advanced search "tool language:Java"
- Sonarcloud[2] → look for projects with coverage value
- StackOverflow[3] → people already posted corresponding project requests

[1] https://github.com/
[2] https://sonarcloud.io/explore/projects
[3] https://stackoverflow.com/

Organization

Project Management

- As soon as the groups are finally arranged, we will create repositories for you on **GitLab**[4] (TUHH-internal)
- Using Git, both your group members and supervisors are able to see the *version history* of your individual commits to follow the overall development of each project
- Final phase submissions are still performed via StudIP

[4] `https://collaborating.tuhh.de`

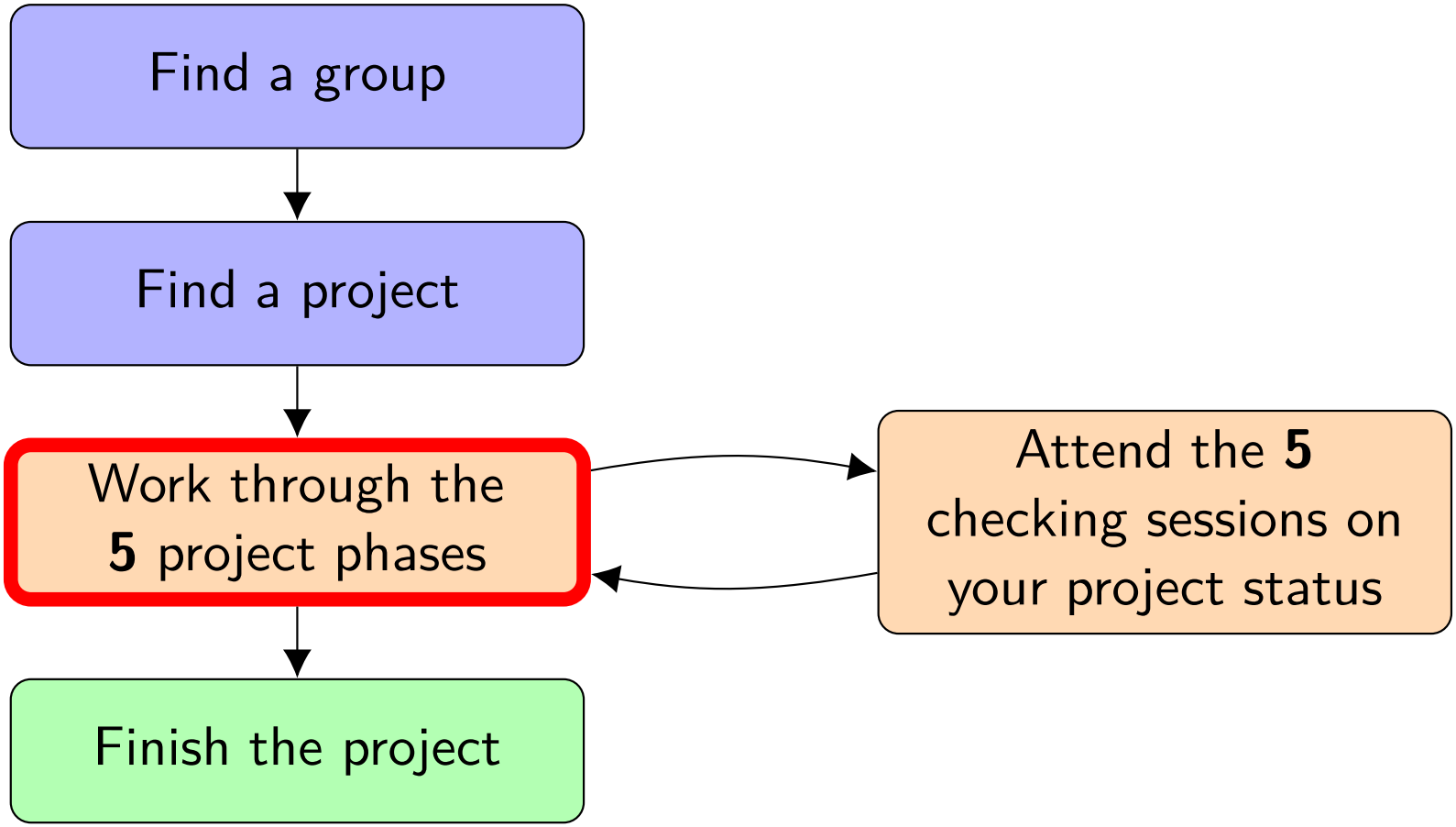
Organization

Project Requirements

The selected project needs to meet the following requirements:

1. The project needs to **consist of multiple classes**, so that each group member can concentrate on one distinct part of the software
2. A **test suite** ($>$ 5 tests) needs to exist alongside with the software
3. The number of **lines of code** (LoC) has to be greater than 10,000 (10k)
4. The software has to be **written in Java**

Project Structure



Organization

Preliminary Schedule (may still be subject to changes)

Lecture			Release		Submission		Exercise Session		Checks
			P0 – Pre (0%)						
(1)	Do, 07.04.22	Intro	P1 - Random (20%)	Do, 07.04.22			(1)	Fr, 08.04.22	P0
(2)	Do, 14.04.22	ISP 1							
(3)	Do, 21.04.22	ISP 2	P2 - ISP (20%)	Do, 21.04.22	P1-1	Mi, 20.04.22	(2)	Fr, 22.04.22	P1
(4)	Do, 28.04.22	Graph 1			P1-2	Mi, 27.04.22	(3)	Fr, 29.04.22	P1
(5)	Do, 05.05.22	Graph 2	P3 - Graph (20%)	Do, 05.05.22	P2-1	Mi, 04.05.22	(4)	Fr, 06.05.22	Res
(6)	Do, 12.05.22	Graph 3			P2-2	Mi, 11.05.22	(5)	Fr, 13.05.22	P2
(7)	Do, 19.05.22	Logic 1			P3-1	Mi, 18.05.22	(6)	Fr, 20.05.22	P2
(8)	Do, 02.06.22	Logic 2	P4 - Logic (20%)	Do, 02.06.22	P3-2	Mi, 01.06.22	(7)	Fr, 03.06.22	P3
(9)	Do, 09.06.22	Logic 3					(8)	Fr, 10.06.22	P3
(10)	Do, 16.06.22	Pres.	Research (20%)		P4-1	Mi, 15.06.22	(9)	Fr, 17.06.22	Res
(11)	Do, 23.06.22	Syntax 1			P4-2	Mi, 22.06.22	(10)	Fr, 24.06.22	P4
(12)	Do, 30.06.22	Syntax 2	P5 - Syntax (20%)	Do, 30.06.22			(11)	Fr, 01.07.22	P4
(13)	Do, 07.07.22	Syntax 3					(12)	Fr, 08.07.22	TDD
(14)	Do, 14.07.22	Guest lect.			P5	Mi, 13.07.22	(13)	Fr, 15.07.22	P5

Organization

Deadline Concept

- There will be **two** deadlines for the project phases P1, P2, P3, and P4
- Handing in for the first deadline allows you to reach the **full amount of points**
- If you encounter major problems with your project, and need a postponement, you can resort to the second deadline
  - You can only reach a **reduced number of points** for that phase in total then

Organization

What is allowed as a group? What is not?

As a group, you **ARE** allowed to ...

- help each other during problem solving and implementation
- work through your individual tasks one after another as a group
- **BUT**: You have to make sure that you **understand your solutions**, and that you are **able to explain every single step** of them during the checking sessions
- Additionally, the commit history of your git repositories will show how much you contributed at each phase

Nevertheless, you **ARE NOT** allowed to ...

- hand in identical solutions (we will both automatically and manually compare your submissions)
- let your team members do the mental work for you (we WILL notice that during the checks)
- submit solutions which are not completely your work

Submission Specifications

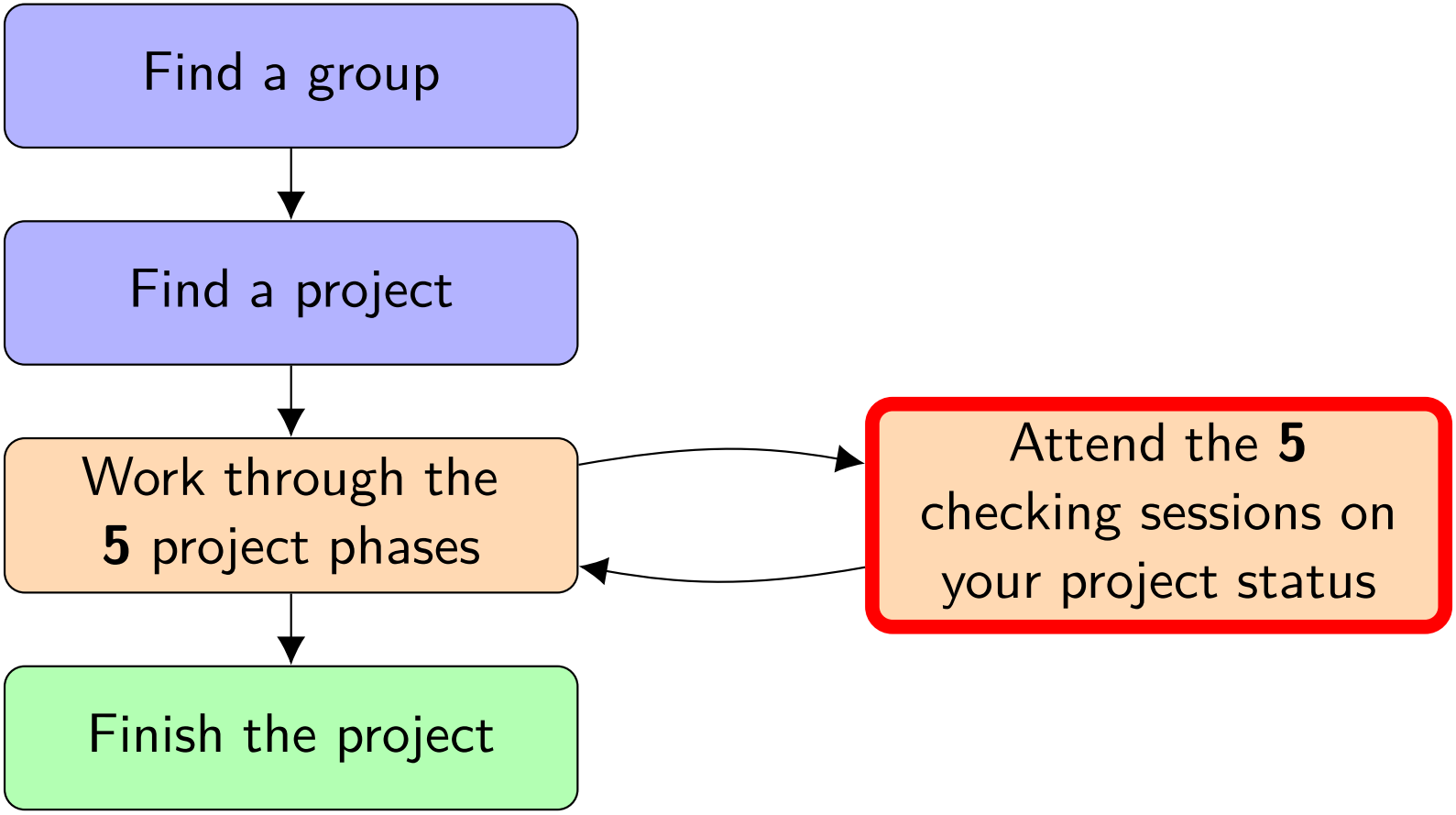
- On each of the task sheets, you will be clearly instructed on the file formats and naming conventions of each submission
- You **HAVE** to follow these conventions (no `*.doc` instead of `*.txt`, no `*.rar` instead of `*.zip`, ...)
- Otherwise, the affected solution **will not be graded** and considered as not handed in

Organization

Additional Track - Research Phase

- In addition to the regular 5 project phases, we offer a *research* phase
- This additional phase will ...
  - expose you to state-of-the-art research
  - allow you to experience the complete cross section of writing, presenting, and discussing research
  - give you an understanding of testing-related topics beyond the scope of the lecture via presentations and problem-based learning
- The tasks of the research phase cover a short ...
  - introduction write-up of an individually selected paper ($450 - 550$ words)
  - in-class presentation on a particular paper topic in groups ($6 - 7$ minutes)
  - paper-related task designed by the groups (an exercise session is dedicated to solving the tasks in-class)

Project Structure



Grading

Structure of the Checking Sessions

- Every group will be checked once during each project phase
- Every member will discuss one (or more) of the following topics with the supervisor (will be chosen randomly):
  1. Problem Identification (to one of the problems you identified)
  2. Decision Making (describe and justify one central decision of the current phase)
  3. Current Knowledge (explain one concept covered in the lecture that is currently relevant)
  4. Open Questions (further describe one open question that you formulated in one of the tasks of each phase)
  5. Software Demonstration (demonstrate the compilation, installation and execution of both your selected software system and the currently relevant tools)

Grading

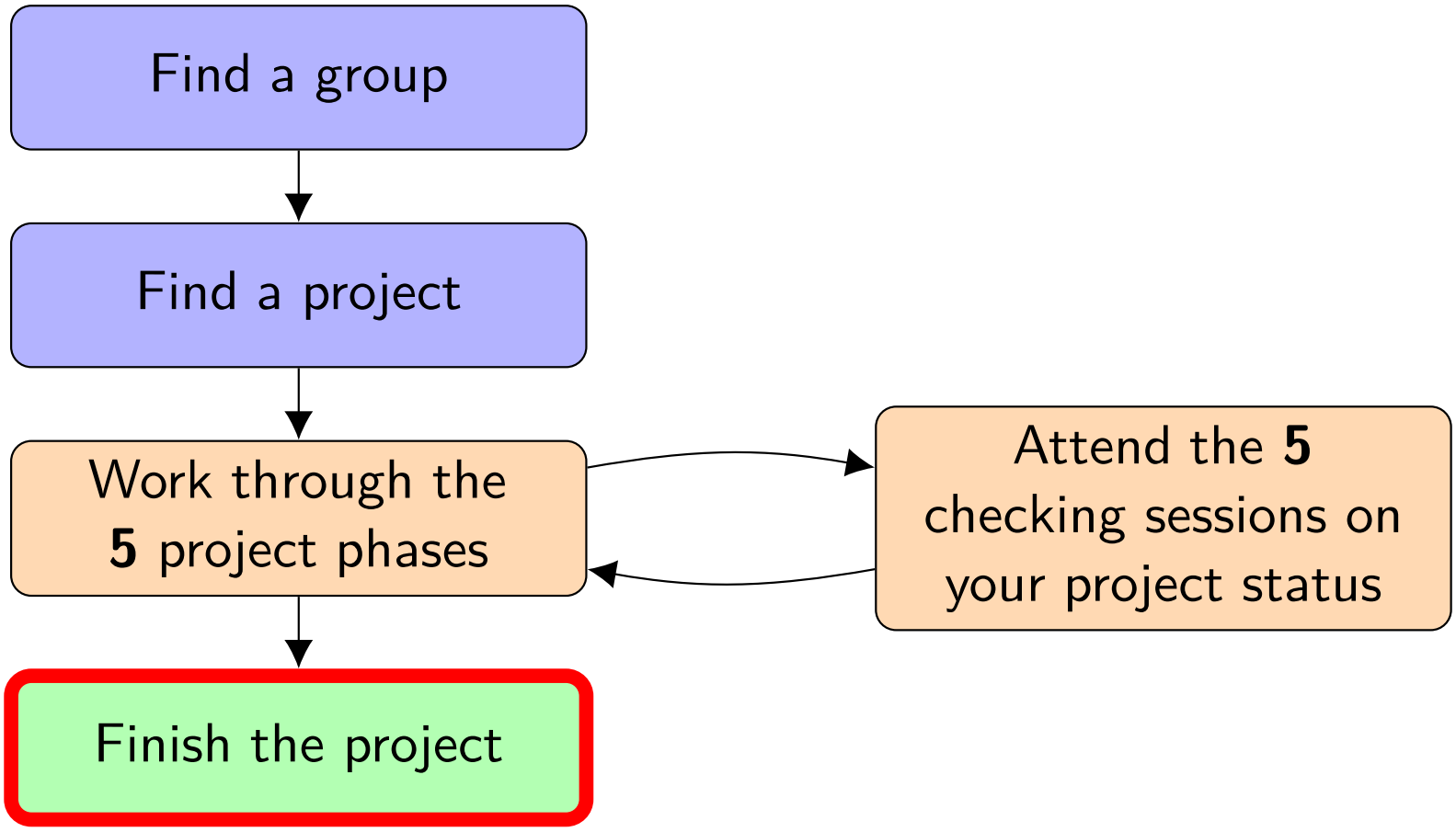
Project Points Plan

Week	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14
Phase 1	rel.	-	20	10	-	-	-	-	-	-	-	-	-	-
Phase 2	-	-	rel.	-	20	10	-	-	-	-	-	-	-	-
Phase 3	-	-	-	-	rel.	-	20	10	-	-	-	-	-	-
Phase 4	-	-	-	-	-	-	-	rel.	-	20	10	-	-	-
Phase 5	-	-	-	-	-	-	-	-	-	-	-	rel.	-	20

The points for each phase will be divided as follows:

- 80% for the submitted solutions
- 20% for the checking sessions
- Note: The solutions for a phase will only be counted if you successfully **participated** in the corresponding **checking session**!

Project Structure



Project Finalization

- Depending on your results, you can consider to contribute to the selected open source project by *making your test suite available*, *indicating a potential bug*, or by *providing a fix* for that bug
- We will try to support you with the undertaking then

Finally...

Things to keep in mind

- Adhere to the deadlines. Solutions will not be graded after the second deadline!
- Your last submission counts!
- Every member submits a solution on his/her own, even when you work as a group.
- Be prepared for each checking category on every meeting $\rightarrow$ No prior information!

Finally...

What do we expect? What can you expect?

Our expectations are that you...

- participate actively throughout the whole project
- are motivated and find a project which you are interested in personally
- stick to the specifications for your submissions
- have fun with the project (and if not, do not hesitate to tell us what is going wrong)

You can expect that you will get...

- continuous support for your work regarding testing related problems
- the possibility to both work in a group or alone for some parts
- feedback at every stage, so you can see if you are moving into the right direction

Questions & Answers