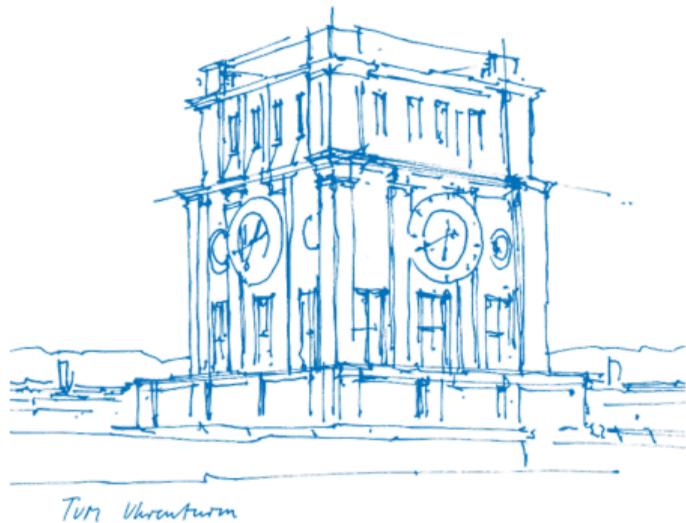


A coherent curriculum track of RSE skills for simulation software

deRSE24

Gerasimos Chourdakis
Technical University of Munich

March 5, 2024



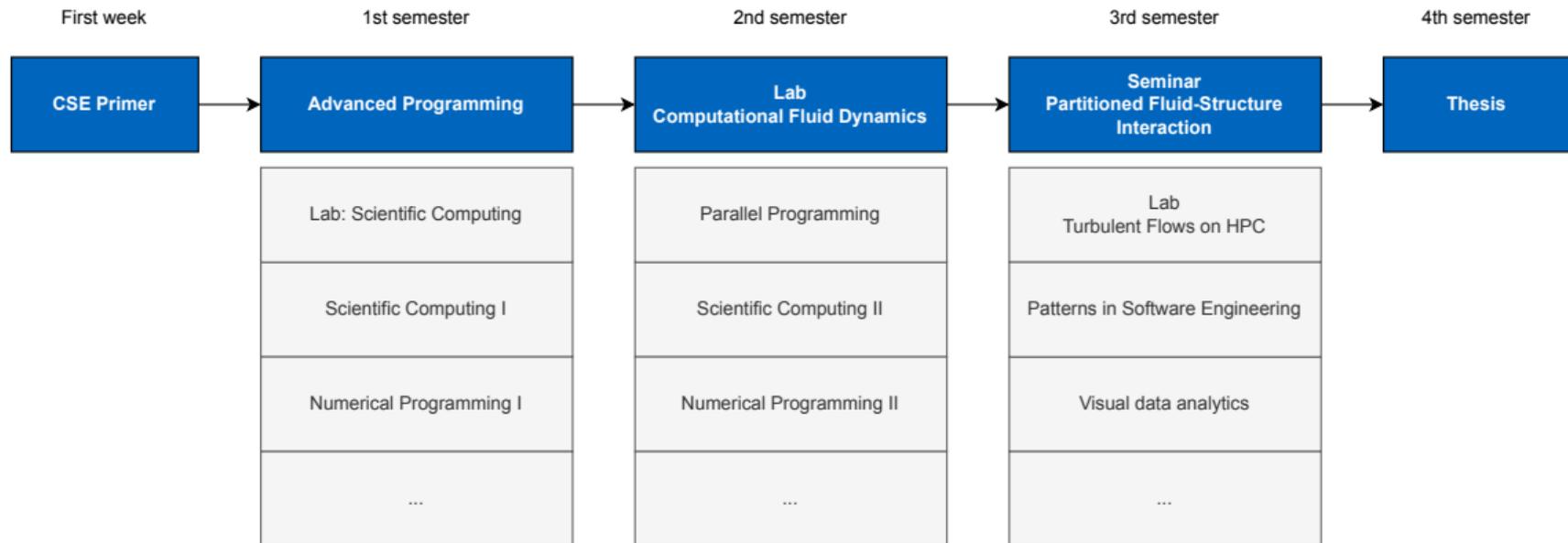
Meet Alex, B.Sc. STEM, CSE newbie



In the M.Sc. Computational Science and Engineering, Alex will learn a lot:

- Numerical analysis
- Computer science
- Applications

Becoming an RSE specialized in simulation software



CSE Primer: Preparing the surface to build upon

	Monday	Tuesday	Wednesday	Thursday	Friday
09:00	CSE regulations	Octave Matlab	C++	C++	CSE Q&A
10:00	Linux				Technical Q&A
11:00					
12:00	Lunch break				
13:00	Working in teams	Project	Project	Project	Project poster session
14:00					Q&A
15:00					
16:00	TUM Welcome event	Pub crawl			Social event
later					

+ sleep!

Advanced Programming (IN1503)

- Starting from modern C++, legacy at the end. (not “from C to C/C++”)
- C++ Core Guidelines, SG20
- Debugger, Profiler, Sanitizers, CMake, Git, ...
- Performance aspects
- Lecture + Tutorial + optional team project

Home Dashboard Help

Material for the WS2023/24 course [Advanced Programming](#) and the corresponding [tutorial](#). Read the [course description](#). We recommend that you participate into the short block course “[C&C Primer](#)” (Moodle) to get a glimpse of basic C++ and Linux tools. Please have a look also in the News for announcements, which you should also receive by e-mail.

News

[Link to the lecture streaming/recording \(ask questions in the chat\)](#)

[Link to virtual tutorial platform \(Tuesdays\)](#)

Organization

Introduction Data types Functions Resource management Build time OOP 1 OOP 2 Templates STL Performance Optimization Closing Guest lecture Bonus project Tools Further resources Exam

Course description

Course format in WS23/24: You will be able to join both the lectures and the tutorials completely online, if you want. The lecture will be in presence with livestreaming (Mondays) will be in presence in MI 02.07.023, the second group (Tuesdays) will be presence-first hybrid, in MI 02.07.023 + streaming/chat. Please see the schedule.

Schedule (WS23/24)

System requirements

Forum: Questions and general discussion

Ask and answer questions here, let's create a community!
To encourage participation, tutors will wait for replies from students before joining the discussion.
You can unsubscribe from this forum to stop getting notifications.

Moodleoverflow: Content-related questions

go.tum.de/897784

Advanced Programming (IN1503): slides

Pass by value -vs- pass by reference

<https://compiler-explorer.com/z/YjP7Pd3W5>

```
int offset(int x){
    int y = x + 40;
    return y;
}

int main(){
    int a = 2;
    int b = offset(a);
    std::cout << a << b << "\n";
}
```

```
void offset(int& x){
    x = x + 40;
    return;
}

int main(){
    int a = 2;
    offset(a);
    std::cout << a << "\n";
}
```

Hands-on: What is the output on the two cases? What is different in the syntax? What is the same?

Advanced Programming (IN1503): slides

TUM

Pass by value -vs- pass by reference

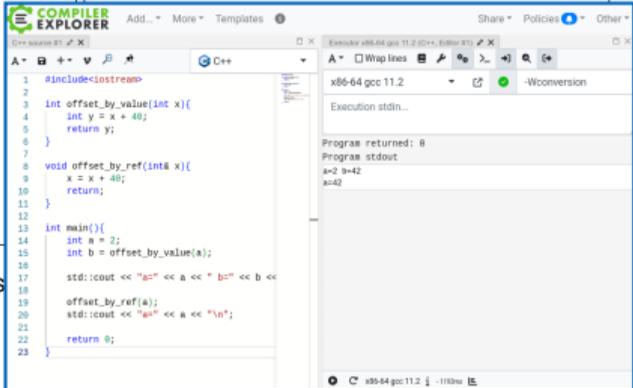
<https://compiler-explorer.com/z/YjP7Pd3W5>

```
int offset(int x){
    int y = x + 40;
    return y;
}

int main(){
    int a = 2;
    int b = offset(a);
    std::cout << a << b << "\n";
}
```

```
void offset(int& x){
    x = x + 40;
}
```

Hands-on: What is the output on the two cases



IN1503 Advanced Programming | 3. Functions | Built on January 26, 2024

3.7

Infrastructure matters!

- Slides made in \LaTeX → versioned in Git → built for different modes on GitLab CI
- Ideas continuously collected in GitLab issues → organized per topic using labels, scheduled
- Student projects on GitLab
- Lecture recordings on TUM Live
- Exam correction & review on TUM Exam

Infrastructure matters!

- Slides made in \LaTeX → versioned in Git → built for different modes on GitLab CI
- Ideas continuously collected in GitLab issues → organized per topic using labels, scheduled
- Student projects on GitLab
- Lecture recordings on TUM Live
- Exam correction & review on TUM Exam

Having at least two instructors accelerates redesigning a course:

Created most of the material in pair programming sessions with Hasan Ashraf in WS20/21.
The one motivates the other to continue and make “Alex” happier.

Infrastructure matters!

The screenshot shows the TUM course catalog interface. On the left, there are navigation tabs for 'Semesters', 'My Courses', and 'Public Courses'. The main area displays a grid of course cards for 'Advanced Programming' (EN1333) across different semesters (October, November, December, January). Each card includes a thumbnail image, the course title, and the start time. The 'Public Courses' section is currently selected.

The screenshot shows the 'Advanced Programming' exam correction page on tumexam.de. The page is titled 'Endterm > Correction' and includes a 'Quick Start' section with instructions: 'Choose which problem (or specifically subproblem) and pass you want to correct. You are then provided the next available exam-waiting which is not corrected yet.' Below this are dropdown menus for 'Problem', 'Subproblem', and 'Correction Pass', along with a 'Start' button. The 'Status' section shows progress bars for four problems, all at 100%. The 'All Problems' table at the bottom provides a summary of exam results.

ERID #	Problem #	Subproblem #	Pass #	Corrected #	Available #	Locked #	Access #	Flags #
E9001	1		1	✓	✓		✓	
E9001	2		1	✓	✓		✓	

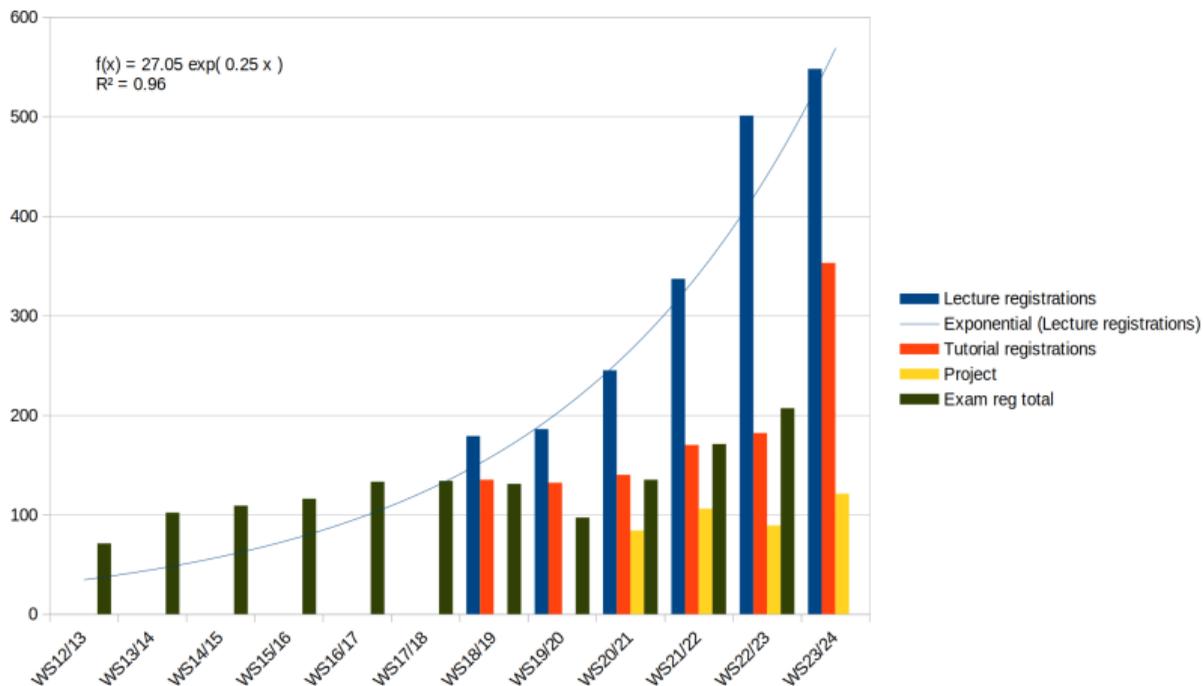
BIG kudos to the TUM RBG and the TUMExam team!

First experience of a team project

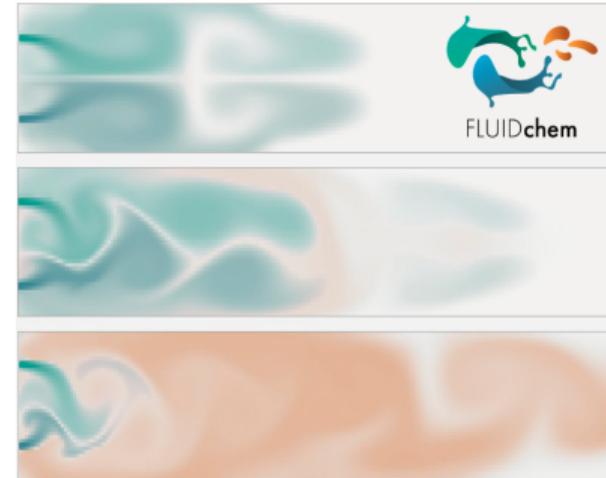
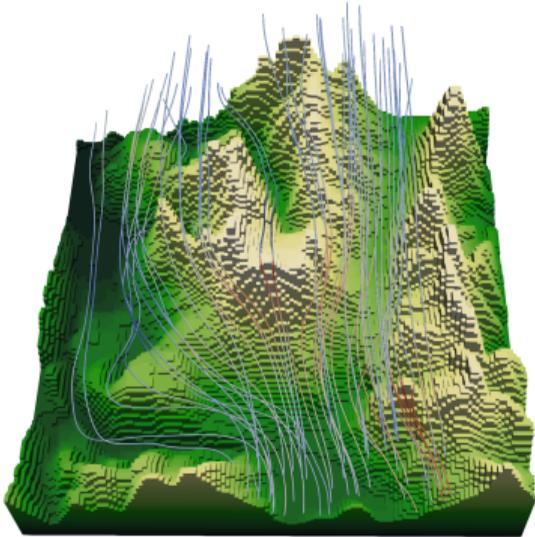
- Must work with someone else, must work on GitLab
- Topic from pool of ideas, or own idea
- Three sprints
- First sprint review by us, next as peer-review
- Moodle Workshop activity + code review
- Do everything → 0.3 grade bonus
- Free to break everything, no grade on the quality

students love it!

The demand grows rapidly



CFD Lab: Working on a longer team project



Examples of final student projects (groups of three)

Challenges and wishes

Challenges:

- Alex avoids Linux, but we need it
- Large, inhomogeneous audience
- Scaling-up peer-reviews
- Option: Contributing to FOSS

Wishes:

- More automation
- Graded assignments
- Distributing exercises via GitLab
- More time for more topics

Planning to publish the material + findings
(license?)

Closing

That was only appetizers – Ask me anything

`github.com/MakisH`

`gerasimos.chourdakis@tum.de`

Advertisement: Join the Munich RSE! Next session: April 11 at the LMU Library.

`de-rse.org/chapter/muc/`