CONFERENCE
WÜRZBURG 2024

Contribution ID: **95** Type: **Talk (15min + 5min)**

A coherent curriculum track of RSE skills for simulation software

Tuesday, March 5, 2024 4:30 PM (20 minutes)

What if your graduate programme actually prepared you with all the software engineering skills you need to participate in the research software community?
The M.Sc. Computational Science and Engineering (CSE) at the Technical University of Munich gathers STEM graduates from all over the world and teaches them elements of numerics, computer science, and applications. CSE graduates are the perfect candidates for developing the next PETSc, OpenFOAM, or Tensorflow. And yet, the programming-related part of the curriculum needed some aligning and dusting.

Over the past six years, we had the opportunity to look at the big picture, redesigning several courses that "tell a story" together. Nowadays, a CSE student can follow a coherent track that prepares them for working as software engineers in an RSE team developing simulation software. We start by preparing the ground with fundamental Linux, Git, Matlab/Octave, C++, and teamwork concepts in the 1-week onboarding course "CSE Primer". In the afternoons of that week, students also work in teams, developing small projects analyzing climate data.

For the rest of the semester, CSE students have to follow "Advanced Programming", a course with an ambitious name which we put a lot of effort in justifying. With the "advanced", we aim to raise the level of the inexperienced, while still offering enough opportunities for already experienced students to grow. The material covers a pragmatic mixture of modern C++ with just enough references of legacy features to be able to work with existing codebases. The slides include code snippets that the students can interact with using the Compiler Explorer. The tutorial exercises include common tools that support the development, including a debugger, sanitizer, formatter, build tools, testing frameworks, and more. An optional project lets students develop their own idea in pairs, or contribute to existing open-source projects, while participating in a code peer-review process. The lectures and tutorial are hybrid, and the in-person exam is supported by TUMExam, a system that offers digital correction and review features. The redesign of this course attracted several students from additional study programs, with the original audience of CSE students now representing less than 10% of the exponentially-growing total audience.

After the first semester, students follow a practical (lab) course. One highlight is the Computaional Fluid Dynamics Lab, in which students work in groups to implement worksheets and their own final project, in a bare-minimum C++ PDE framework, receiving code reviews on GitLab and maving their first steps towards parallel programming and performance optimization. Cross-references between Advanced Programming and CFD Lab make the two courses coherent, without discouraging external students to join. The (not offered anymore due to staff shortage) seminar Partitioned Fluid-Structure Interaction lets students expand their research skills specific to CFD, writing their own paper and participating in peer-reviews.

This talk will give an overview of these courses, discussing several didactical and technical elements applied in each, concluding with not-so-obvious good practices.

Slot length

Primary author: CHOURDAKIS, Gerasimos (Technical University of Munich)

Presenter: CHOURDAKIS, Gerasimos (Technical University of Munich)

Session Classification: Teaching RSE Skills

Track Classification: RSE Research and Communities: Teaching RSE Skills