

# Introduction

It is common for students learning programming to go through courses in a practical way with assignments and programming exercises. Assignments and exercises usually consists of a number of tasks that the student will have to solve. When the student is finished, the code is submitted to the teacher. The teacher downloads the code, runs it and grades the student. This approach is time consuming, and by the time the teacher has graded all the exercises, the students have already started on their next assignment. This can lead to students not fully understanding the curriculum and in worst case miss out on key parts of the course. When the courses get more advanced, and the code base increases, this problem gets even bigger. A lot of time is wasted on the overhead work, rather than giving student the help and feedback they need. It also requires a lot of work for the teachers, and can often result in students being left behind by the teacher because of pressure to get all the grading done. Even though the system discussed is in use now, it does not mean that it cannot be changed and improved. One way of solving this problem is by using a system of automatic testing and grading. Such a system is already implemented on other universities, but these systems usually lack key features. With that in mind, former master student Heine Furubotten created a system call Autograder. Autograder is a system for automatic testing, code reviewing and grading. The system collects student submitted code from Github, a version control system, and grades it based on test written by the teacher. The system has already been in use on some of the courses held at bachelor and master level at UiS. Autograder requires some basic knowledge of Github and Git, however, these technologies will have to be taught at some point later on anyway. The process of submitting code is easy, as well as the process of getting feedback.

Autograder works the same way many programs in the industry work, and mimics the the real world scenarios that the student will encounter later. When the students graduate and start working in the industry, a similar process of code review and test-driven development will probably be the most common way of programming. It is therefore not an issue to rely on the students learning version control system. The system has already been used on some of the advanced programming courses at UiS, and has proven

a valuable asset. The system worked as expected and made the job of the teachers easier, by showing all the student in a web-based system and their progress. Automatic testing was done though code submitted to Github. The code was downloaded and run on computers at UiS, showing the results and build log in the browser. This made it easy for both students and teachers to monitor progress, and evaluate the work. Feedback was given along the way, and helped improved the system with direct feedback from the users. According to Heine, the system improved how the assignments and grading process is done. It also made it simpler for the students to focus on the tasks at hand, rather than having to think about a cumbersome and slow grading process.

Automatic testing of code and grading is not a new idea. Many similar systems are in use at other universities. The biggest difference is that they rely on third party software for testing, and can in many cases not fit the need of the teachers and students. The Autograder system is developed at UiS with direct student and teacher feedback. This gives an advantage when it comes to implementation of useful functions, and is therefore tailored for the needs of the advanced programming courses at UiS. Nevertheless, this does not limit the functionality of the program, which can be implemented on other universities. Autograder is not limited to programming languages design-wise, however, it is mostly tested with the Go programming language. Both student and teachers have been very pleased with the system, and now that it has been used for some time, relevant user-data can be gathered to improve the system even more.