Background

Introduction to the problem

Computer science has become ever more prevalent in schools. With the increase in the amount its being taught, tools have been developed to increase the ease of teaching it to students. Both the GCSE and the A level computing examinations require familiarity with a programming language and the ability to be able to implement algorithms in code. However, computer science is unlike other subjects taught at schools; the solutions to programming tasks set to students are often submitted to teachers in the form of either source code or compiled executables.

This presents a problem to computer science teachers in terms of marking the work. The current method of marking a student’s work is as follows:

1. It is an intensive process to launch each student’s work
2. Manually enter the test data
3. Record the output of the program
4. Assign a mark based on the performance of the program
5. Provide feedback to the student based on coding style and other factors

This then needs to be repeated for each student in the class. Inevitably, the larger the class being taught, the less time the teacher will have to customise the feedback for each student.

Current solutions and issues

One notable example is codecademy. This is an online web app that teaches pupils several languages through interactive guides. These guides provide a quick and easy way to familiarise students with the idea of data structures and control flow in programming. However, this is tool does not effectively work in the classroom. The format of the teaching is very rigid and does not allow for teachers to come up with tasks to set, rather only choose from pre-existing exercises.

Other alternatives include unit tests. These are programs that allow for the automated testing of functions and methods in a given program. However, they can be complex to set up, for example installing new modules when using Python, and tedious for students to write test cases. This often results in students either writing minimal test cases, or none at all.

Analysis of interview with client

The client for this project is a computer science teacher at my school. During the interview I asked several questions related to the current process of marking, what they thought would be good to have in an automated marking system, and what would be essential in an automated system. The client commented on the fact that Codecademy is quite good at teaching beginners how to code, but could be somewhat repetitive. They also said that they were interested in the ‘gamification’ of either the marking process or the homework itself. In terms of necessities, it would have to be trivial for students to upload their work, regardless of their location. Furthermore, it would have to be easy for them to see all submitted work themselves. In terms of the marking process, it would have to be able to handle code that may not have exact function names and match these to the functions that are being expected.

Requirements of the project

1. Easy to access and use interface
2. Easy to create and design tests and expected results for assignments
3. Ability for teacher to view all students’ work along with results of testing
4. Students get customised automated feedback on submitted work
5. Gamified element for the assignments

Project specification