

# Web Platform for Peer

## The Problem

Many Computer Science courses use programming coursework assignments to teach students.

Ideally, in order to give useful feedback these should be fully tested, but it is impractical for teachers to do so on large class sizes, and relying on this increases the turnaround time for feedback.

Research has shown that activities like peer assessment offer many benefits to students, particularly when it is done anonymously.

This project created a web platform offering a peer-testing approach

# er-Testing

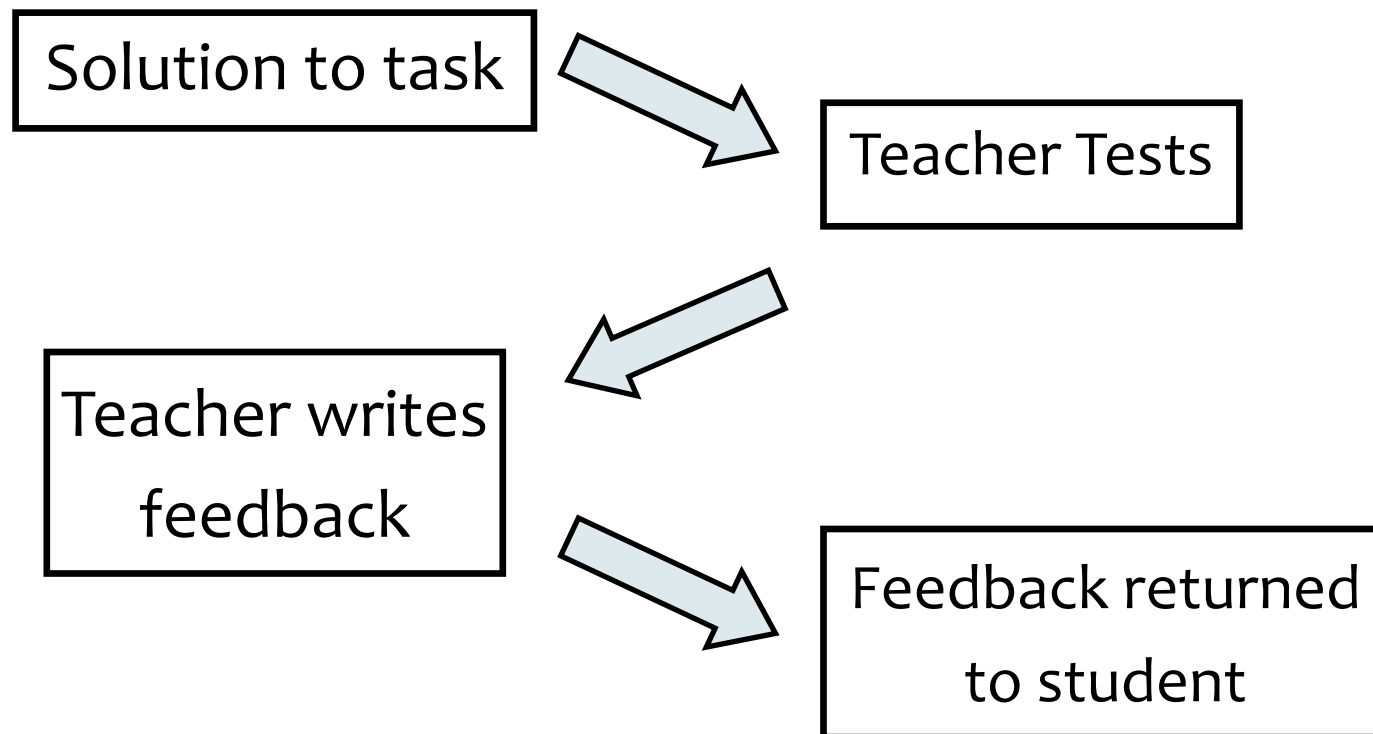


ments to

t it is  
reduces

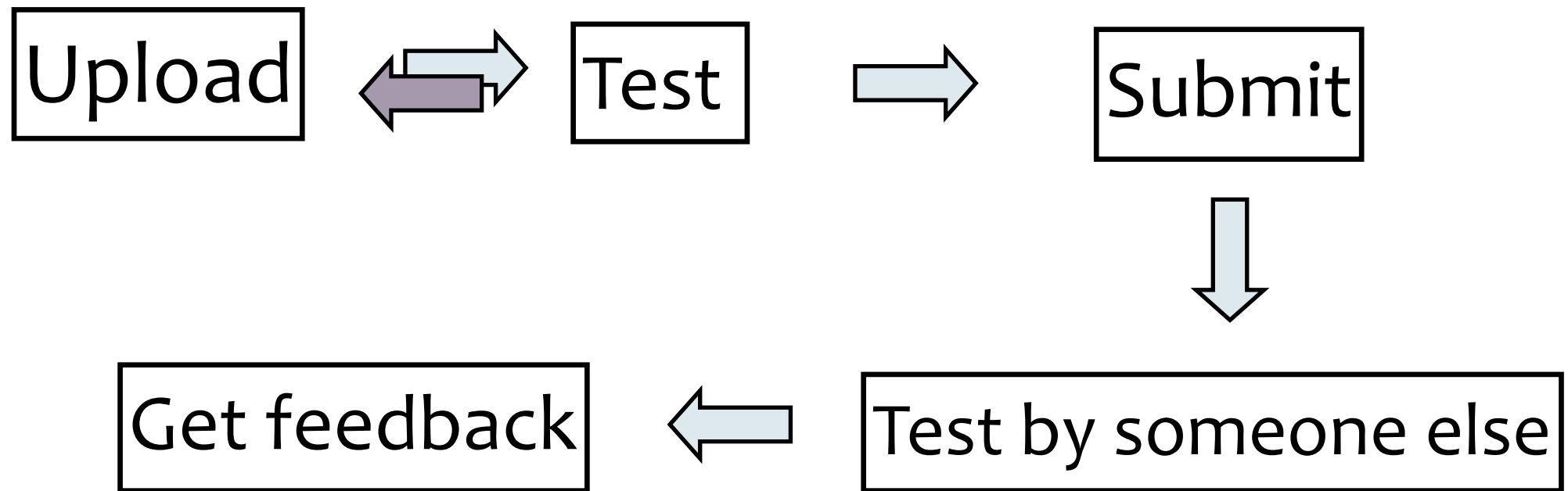
efits to

to giving



The teacher needs to test and give feedback on each solution

feedback on programming coursework assignments.



Students complete the assignment and are (anonymously) given another student's solution to test and give feedback on

for every student on the course, which can take a long time

## The Solution

Using Python and the Django web framework I have created a website that allows students to complete a large portion of the coursework assignment in one place.

The site can serve coursework assignments and allows a user to:

- Upload their solution to assignment
- See if it matches the basic interface required for testing
- Run their own test cases against their solution
- Run their own test cases against a reference solution
- See other students solutions and test those
- Give feedback to other students for their solutions

The coursework tasks involved use python 3 code and the testing is written using the python unit test module.

# Evaluation Study & Results

To test the usability, correctness and usefulness of the website, student participants of Computer Science courses in Edinburgh and Dubai used the website.

The participants in the study completed two tasks (implementing Quicksort and Binary Tree), and participants alternated using the website to complete tasks & peer-testing, and peer-testing through email.

This study revealed that the website worked correctly, but had some usability issues.

The website successfully allowed the students to upload their files and view them on both their own and each others solutions. However, due to mis-expectations of how the website worked, some students were unable to view files before being tested when writing feedback.

The anonymity offered by the website worked well, people were prevented from seeing who wrote code and feedback, and only 5 participants thought they might know who they were writing feedback for.

Regarding the running of the tests, students noted that the website was

the

ort and a  
the task

ability

run tests  
lanations  
eing

nted  
ught they

S

```
1 import unittest
2 from tree import Tree
3
4 class Test(unittest.TestCase):
5
6     #fullbalancedtree = [10, 5, 15, 3, 7, 12, 17, 1, 4, 6, 8, 11, 13, 16, 20]
7
8     def test_newTree(self):
9         t = Tree(10)
10        self.assertIsInstance(t, Tree)
11        self.assertEqual(0, t.find(10), "Unable to find root value")
12
13    def test_recursion_limit(self):
14        t = Tree(0)
15        for i in range(1, 1010):
16            try:
17                self.assertTrue(t.insert(i), "Python recursion limit reached during insertio")
18                self.assertEqual(t.find(i), i, "Python recursion limit reached during find")
19            except RuntimeError:
20                self.fail("Recursion limit reached")
21
22    def test_repeated_insert(self):
23        t = Tree(10)
24        for val in [5, 15, 3, 7, 12, 17, 1, 4, 6, 8, 11, 13, 16, 20]:
25            self.assertTrue(t.insert(val))
26            self.assertFalse(t.insert(val), "Didn't return false on re-insertion of existing")
27
28    def test_depth(self):
29        t = Tree(10)
```

« *helpful* » but « *not as quick as bashing something out on the command line* ».

When asked, the participants felt that such a website would be of benefit to current Computer Science courses, in particular modules such as « Data Structures & Algorithms », and that the cross-campus interaction offered was « *interesting* ».

## Future Work

Investigation could be made into integrating an anonymous style discussion system as a replacement for the one comment of feedback currently allowed.

Also, the existing university VLE could be used to provide the file upload and grading backend.

Currently, only python exercises are supported. Additional language and testing types could be developed.

ne ».

fit to the

d was

```
29     t = Tree(10)
30     for val in [(5,1), (15,1), (3,2), (7,2), (12,2), (17,2), (1,3), (4,3), (6,3), (8,3),
31               ]:
32         t.insert(val[0])
33         self.assertEqual(t.find(val[0]), val[1], "Incorrect depth returned")
34         self.assertEqual(t.find(10), 0, "Incorrect depth returned for root node")
35
36     def test_failed_find(self):
37         t = Tree(10)
38         for val in [5, 15, 3, 7, 12, 17, 1, 4, 6, 8, 11, 13, 16, 20]:
39             t.insert(val)
40             self.assertEqual(t.find(45), -1, "Incorrect return on find failure")
41
42     def test_negative_int(self):
43         t = Tree(-5)
44         self.assertTrue(t.insert(-4))
45         self.assertTrue(t.insert(-6))
46         self.assertTrue(t.insert(-3))
47         self.assertTrue(t.insert(-7))
48         self.assertTrue(t.insert(10))
49         self.assertEqual(t.find(-5), 0)
50         self.assertEqual(t.find(-4), 1)
51         self.assertEqual(t.find(-6), 1)
52         self.assertEqual(t.find(-3), 2)
53         self.assertEqual(t.find(-7), 2)
54         self.assertEqual(t.find(10), 2)
```

#### Feedback

As you should be able to see, your solution failed all of the tests that I ran on it. One major reason for this is down to a few indentation errors. Though I think you're aware of it, I should say in case you're not that in Python indentation is semantic, that is it has meaning. Where Java, C et al. use braces ({} to denote the start and end of blocks, Python uses indentation. The error is that all of the methods declared after `__init__` should be indented one more level, otherwise (as in the file) they are not considered part of the class `Tree` as required by the specification. The best way to avoid these errors is to use an IDE with auto-indenting, or something like

The feedback view of the website, showing all of the files involved in the testing