One Two Test Experiment

When the participants arrives

Consent forms

Give each participant a consent form to read and sign.

Explain what will happen in the experiment

As you know, this is a research project on problem solving in different programming languages. In this experiment, you will work on two problems. The first one is a "hello world" warm up problem. All you need to do is get the program to print "Hello World". It's a warm up problem because we want you to get comfortable running the automated unit tests that we will use to verify that your solution is correct. Once you've finished the hello world problem, let us know, and we'll let you move on to the second problem, called "the saddle points problem." The saddle points problem is done in two parts. After you finish part 1, let one of us know, and we will come over and get you started on part 2.

In this experiment, we are interested in how easily people can pick up and continue code that was written by someone else. So that means we want you to use informative variable names and include comments wherever you think it might be helpful for someone else reading your code.

You should also know that it's ok to use the internet to Google things, but please don't try to find exact solutions to the problem you are solving. It will be really easy for us to tell if you've just copied and pasted in an entire solution you found some-

where else on the web, but if you need to look up the docs for a builtin function or something, go right ahead.

Now we will get you set up at your computer, show you how to run the tests, and let you get started on the first problem. Once you've finished the first problem, we will come over and you can show us that you can run the tests and that the tests pass. Then you can start on the second problem.

Show each participant how to run the tests

Java: Eclipse

Run the tests by opening the test file and pressing the green "Play" button in the menu bar. Run the file as a JUnit test.

OR: run the tests from the command line by navigating to the problem directory and running `gradle test`:

```
cd ~/problems-java/saddle-points/
gradle test
```

Python: PyCharm

Run the tests by opening the test file and selecting "Run 'pytest for test_file_name.py'".

OR: run the tests from the command line by navigating to the problem directory and running `pytest`:

```
cd ~/problems-python/saddle-points/
pytest
```