

University of Liverpool Lab test assessment

COMP285/COMP220 Lab test (Ant and Junit)

Date April 2019

University of Liverpool

Computer Science

This is an open book test but you must not use email or communicate with any other students in the course of this test, but you can use any of the subject resources on Vital. This test sheet if in hard copy should be handed back to the invigilator at the end of the test.

It is recommended (but you don't have to) use Eclipse to edit the source files.

All the files are in the src directory....

For this test you must produce a build directory and ant build file.

You must also produce a Junit java test file call RegistrationHelperTest.java with tests to test the code found here. A stub form of this test file is provided for you, please remember to change the method **getStudentID()** so that it returns your 8-digit student id.

There are 2 files, RegistrationHelper.java and RegistrationHelperTest.java

NOTE The RegistrationHelper java you have been contains **no bugs**. So all tests should pass.

RegistrationHelper.java is the file you will be testing and it should be copied into the src directory.

IMPORTANT The package name for all the files in this project needs to be labtest, so please copy the files into the correct source directories. So RegistrationHelper.java needs to be in src\labtest

Here is the specification of the code.

There is 1 method called boolean checkUsernamePassword(String username, String password)

This method will return true if both the username and password are valid, if the username or password are invalid it will return false.

The following are the validation rules:

1. If either the username or password are null then the method will return false.
2. The username must be at least 8 characters long and must start with an alphabetic character (A-Z, a-z).
3. The username must be no more than 12 characters long.
4. The password must be last least 8 characters long.
5. The password must contain at least one lower case letter.
6. The password must contain at least one upper case letter.
7. The password must at least one a digit.
8. The password must have a special character one of the following ! " £ \$ % ^ & * ()

The test source files for this lab test should be stored in test\src.

There should be a directory called build\classes where the classes are stored. **(Please turn over)**

University of Liverpool Lab test assessment

Your ant file should compile Main.java and RegistrationHelperTest.java and then leave the result in build\classes.

Submission

Your submission must consist of the following

A zipped up file which is the whole build directory, the name of this file should be named

LabtestXXXXXX.zip where XXXXXXX is your long University id number.

This zipped file must contains.

A build.xml file which compiles makes output directories if needed, copies files from test\targetClasses to build\classes, builds the test code and runs the JUnit tests and also produces reports about the tests in XML and HTML.

The directory structure should have the following directories

build\classes Containing all classes for the application (test and target classes)

test\reports Stores HTML and XML reports

test\src Test code containing the code **RegistrationHelperTest.java**

src Containing the files **Main.java** and **RegistrationHelper.java**

lib Any libraries you may need e.g. JUnit

Marking

The marks are assigned as follows:

Quality of the JUnit file	50%	broken down as
Appropriate structure of file and readability	10%	
Ability to reveal bugs	40%	

Quality of Ant file	50%	broken down as
Production of correct final outputs	25%	
Formatting, readability, good use of properties	25%	

Note for the “Ability to reveal bugs” part of your assessment you tests must do the following:

1. Run with no failure if the code has no bugs, if your tests fail for the bug free code, then you will get **zero marks** for this component.
2. Reveal bugs if the code has bugs in it, your tests will be tested against various versions of the code with bugs that have been added on purpose

TIPS Make each assertion test case test only 1 issue at a time. Check all boundaries, edge cases.