ISTANBUL TECHNICAL UNIVERSITY
FACULTY OF COMPUTER AND INFORMATICS ENGINEERING
COMPUTER ENGINEERING DEPARTMENT

BLG475E - Software Quality and Testing
2019 -2020 Spring
TAKE-HOME FINAL EXAM

Release Date:	30.06.2020 12:00
Submission Date:	01.07.2020 14:30
Duration:	26.5 hours
Instructors:	Dr. Ayşe Tosun

Declaration of authorship

Name, Surname:	(Please insert your full name with your hand-writing here)
Student ID:	(Write your student ID here)

Please tick to confirm the following:

I have read and understood the University's disciplinary regulations concerning conduct in examinations and, in particular, the regulations on plagiarism.	□
The answers I am submitting is entirely my own work except where otherwise indicated.	□
I have clearly indicated the presence of all material I have quoted from other sources, including any diagrams, charts, tables or graphs.	□
I have clearly indicated the presence of all paraphrased material with appropriate references.	□
I have not copied from the work of any other person.	□

Date and Signature:	(Please insert your hand-written signature here)

Before you start, read carefully:

1. This final exam is designed as a take home with open books, open slides. You are allowed to use online resources by providing appropriate references to the quoted or paraphrased material. All questions need to be answered individually; no group work is allowed. Please sign the declaration of authorship at the first page before submitting your report.

2. The questions in this take home exam are related to the web application tested in the third homework. Please follow the instructions in every question regarding the page limits and suggested reading materials.

3. All the reports will be in PDF and submitted to Ninova. Please remember that there will be multiple exams during a day at the university, and all of the exams will use Ninova for declaring the questions and collecting the answers. Therefore, there might be delays with your submissions, or network issues. Please consider multiple minute-delays during the submissions. If you encounter any problems related to your home's wireless connection, please use personal hotspots from your mobile phones if possible. They might give a better connectivity during submission.

4. If you have difficulties while submitting your report to Ninova, you can send your report to tosunay@itu.edu.tr until 1st July 14:50 with your explanation.

5. If you have any questions related to the tasks announced in this take home, please send an email to tosunay@itu.edu.tr and bakkoca@itu.edu.tr before 1st July 10:00.

6. If you have additional issues (issues affecting your attendance to the exam or your submission, etc.), please send an email to tosunay@itu.edu.tr.

In the third homework, you have learned to automatically test a web based application (migros.com.tr) for a given scenario considering both positive and negative cases. The scenario was already provided to you, and your job as a tester was to check whether the system correctly performs according to the specified requirement. The questions below will guide you to generate other test scenarios with multiple inputs. You are not expected to execute these scenarios on the application.

1 (30 pts). List **at least 12 tasks** that can be tested on this web application. Consider that each task corresponds to a user story/action, and that it is atomic. You can name each task by yourself and you must **write a short explanation** about what is expected in this task in terms of user actions. For example,
login is a task and its description is: "a user logs into the system with his/her phone number and a code received via SMS".

The tasks can be related to or dependent on each other. **Identify these relationships** and **depict** these tasks' relationships **visually** in a **tree-like model.** The relationships can be defined with the following attributes in your model:

precedes e.g. **login** must precede **addToBasket**
succeeds e.g. **selectDeliveryTime** must succeed **selectDeliveryAddress**
repeats e.g. **searchProduct** can repeat multiple times in a user scenario.

Write your answer below this question using at most TWO PAGES.

2 (20 pts). Generate a test scenario using the model that you created in the first question. **Check the node coverage in your model** and create your scenario with **as high coverage as possible**.

Your test scenario must be different from the two scenarios provided in the third homework. Furthermore, your test scenario must contain a task with *repeats* attribute at least once.

Remember that, due to the repeating task you will choose, your scenario can have several variations of execution. For example, assume that your scenario is **login, searchProduct, logout**, and it includes **searchProduct** as the repeating task. This scenario can execute in the following forms:
login, searchProduct, logout
login, searchProduct, searchProduct, logout
login, searchProduct, searchProduct, searchProduct, ..., logout.
This can go on infinitely due to the property of the **searchProduct** task.

Show how your scenario can also have several variations due to the repeating tasks. **Propose two strategies to limit these variations** while testing the scenario.

Resources: Check the slides for Structural Testing, specifically path testing, path coverage, boundary interior path coverage.

Write your answer below this question using at most TWO PAGES.

3 (25 pts). The same problem (infinitely many variations of a test scenario) also occurs while generating test inputs. For example, the same scenario with different inputs is provided in the third homework, i.e., with “kırmızı et” and “bebek bezi”. Suppose you will test the following scenario:
login, searchProduct, selectAmount, addBasket.

But this time, you will only search “fruits” that can be purchased in grams or kilograms via migros.com.tr. So your search task would be limited to all the items in this category, and their amounts can differ.

Propose a method to cluster test inputs in such a way that only a few samples from each cluster would suffice to test the mentioned scenario. Then, **generate test inputs selected from these clusters** to test the scenario above.

Resources: Check the slides for Testing Types and Strategies. Pick the method suitable for this context and explain why.

Write your answer below this question using at most ONE PAGE.

4 (25 pts). A tester’s responsibility is to execute the tests using the defined scenarios and associated inputs, and to report the issues/bugs to the developers. The issues are later fixed by the developers, and testers have to re-run the scenarios. This process is called regression testing. Due to limited time, testers need to filter out or prioritize some of the scenarios and test inputs during regression testing.

Assume that you are a tester in the regression testing team of migros.com.tr, and your task is to find the **most effective (fault-finding)** test scenarios to re-test the system after bug fixes.

From all the scenarios that you can generate based on your model in the first question, **which ones will you re-run? Why?** Give **examples**, and **explain in detail.**

Resources: Explain your rationale by referring to the regression test selection and prioritization approaches in Testing Types and Strategies, and fault-finding test inputs in Fault-based Testing.

Write your answer below this question using at most TWO PAGES.

5 (BONUS 40 pts). Use the test scenario you proposed in the second question, and choose test inputs using the method you proposed in the third question. Implement this test scenario and execute it on migros.com.tr. You can pass the login action by manually entering the login credentials during the test execution. Upload your project to Google Drive/Dropbox/GitHub and add the link to your report. You will be called for a demo if you implement this part.