UNIVERSITY
OF SKÖVDE

School of Informatics

WRITTEN EXAMINATION

Course Software Testing G1F, 7.5hp

Sub-course

Course code IT373G

Credits for written examination 5hp

Date 2025-10-24

Examination time 14:15-19:30

Examination responsible

Teachers concerned

Aid at the exam/appendices

Other

Instructions

- ☐ Take a new sheet of paper for each teacher.
- ☒ Take a new sheet of paper when starting a new question.
- ☒ Write only on one side of the paper.
- ☒ Write your name and personal ID No. on all pages you hand in.
- ☒ Use page numbering.
- ☒ Don´t use a red pen.
- ☒ Mark answered questions with a cross on the cover sheet.

Grade points

Examination results should be made public within 18 working days

Good luck!

Total number of pages: 3

Grading

This exam contains five sections. Each section examines the student with respect to one or two of the learning goals specified in the course plan. In order to pass the exam, a student must pass all five sections during that (single) exam. For your convenience, each section lists the relevant examination criteria.

==========

Section 1

Examination criteria: (i) The student can describe the process for model-driven test design (mdtd) as well as its different activities, and (ii) The student can describe the advantage of agile approaches such as test-driven development (tdd) and briefly describe the tdd approach.

Question 1

MDTD allows one test engineer to do the math and leaves the rest of the activities (such as finding test values, automation or execution) to traditional testers and programmers. For each of the below artefacts developed in an MDTD process, **describe** and **exemplify** what the test engineer does to create it.

a. Model/structure
b. Test requirements
c. Refined test requirements/test specification

Question 2

The TDD approach is an agile test-first software development model with both advantages and drawbacks. Describe at least three advantages and two drawbacks.

The level of detail and number of arguments determine the grade.

==========

Section 2

Examination criterion: The student can explain the limitations of software testing

Question 3

What are the main limitations of SW testing, and – given these limitations – why is it still important to design test sets with high quality? ***Note****: This is a question with many possible relevant answers, so you should elaborate on your answer and give at least three arguments.*

The number of arguments and examples determines the grade.

==========

Section 3

Examination criterion: The student can explain the given test techniques in sufficient detail

Question 4

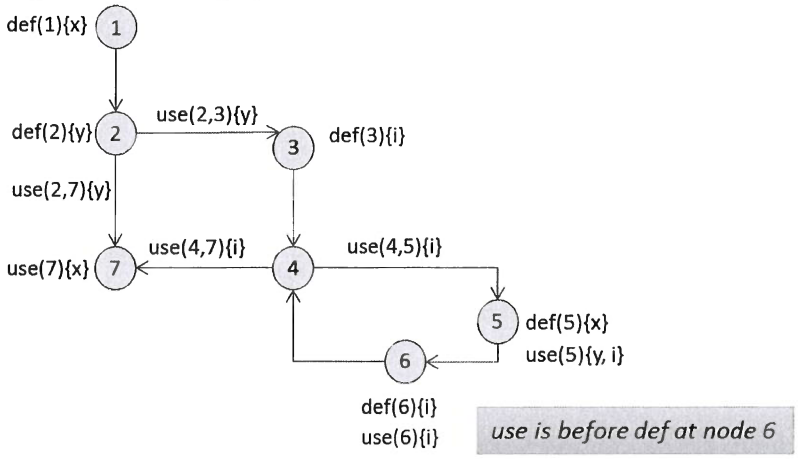
Correlated active clause coverage (CACC):

1. Describe the technique (i.e., how it works)
2. List the CACC test requirements for predicate P (i.e., required rows) based on the table below. **Give all feasible alternatives for each clause.**

Row	*A*	*B*	*C*	*Predicate P*	P_A	P_B	P_C
1	T	T	T	T	T	F	T
2	T	T	F	F	T	F	T
3	T	F	T	T	T	F	T
4	T	F	F	F	F	F	T
5	F	T	T	F	T	F	T
6	F	T	F	T	T	T	T
7	F	F	T	F	T	F	F
8	F	F	F	F	F	T	F

Question 5

2. All-use coverage
   a. Describe the technique (i.e., how it works)
   b. List all the du-paths and indicate in this list which of these du-paths you need for all-use coverage of the below graph



Question 6

Mutation, the ROR mutation operator

1. Describe the ROR operator with your own words (i.e., how it works)
2. List the ROR mutants for the given code fragment

```
if ((m4 != 0) || ((m100 == 0) && (m400 != 0)))
   daysIn[2] = 28;
else
   daysIn[2] = 29;
   numDays = day2 + (daysIn[month1] - day1);
for (int i = month1 + 1; i <= month2-1; i++)
   numDays = daysIn[i] + numDays;
```

Question 7

For a higher grade, also: Compare and contrast edge-pair coverage and prime-path coverage. Your comparison should focus on (*i*) the effectiveness with respect to the probability of exposing failures, (ii) the cost with respect to the number of test requirements, and (*iii*) the usability with respect to infeasible test requirements.

==========

Section 4

Examination criteria: (i) The student can explain the common concepts in software testing and test automation, and (ii) The student can describe the main functionality given by a test automation framework such as JUnit.

Question 8

Explain the following concepts: *test coverage criterion, test case* and *test suite.* In addition, use an example to describe how these three concepts are related.

Question 9

Explain the three concepts: *fault*, *error* and *failure*. Make sure that your explanation is detailed enough to distinguish them.

Question 10

In the JUnit framework, what is an oracle, and how is it used to determine the result?

Level of detail determines the grade.

==========

Section 5

Examination criterion: The student can give at least one very good argument for why the use of coverage criteria helps testers get high-quality tests.

Question 11

In this course, we have discussed several advantages of using coverage criteria. Two of the advantages concern the quality of the test set and test maintenance. How does the use of coverage criteria help the tester improve the test set, and how does it make it easier to maintain the test set?
The number of arguments and the level of detail determine the grade.