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 2024-12-13   Examination time 14:15-19:30

Examination responsible

Teachers concerned   András Márki

Aid at the exam/appendices   Birgitta Lindström

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!

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, the student therefore, needs to pass all five sections. For your convenience, each section list 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

Explain why model-driven test design makes the activity of test design easier and also how the mdtd approach supports giving responsibility for different testing activities to testers with different skills and background.

Question 2

What are the main advantages with an agile development process such as tdd? Name at least three advantages.

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

==

Section 2

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

Question 3

Assuming that you have a test process with high quality, you are using the strongest of criteria and you no longer find any faults in the system – why is it that you still cannot guarantee that the software is correct? Note: This is a question with many answers so you should elaborate on this and give at least three arguments.

Number of arguments and examples determine the grade

==

Section 3

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

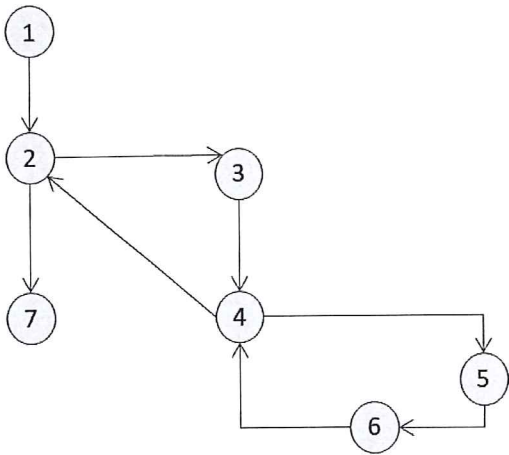### Question 4

Correlated active clause coverage (RICC):

1. Describe the technique with your own words (i.e., how it works)
2. List the RICC 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

Prime paths coverage:

1. Describe the technique with your own words (i.e., how it works)
2. List the test requirements for prime-paths coverage of the below graph (i.e., the required prime paths)

Question 6

Mutation, the ROR mutation operator

1. Describe the ROR operator with your own words (i.e., how it works)
2. List all 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 all-defs coverage and all-uses coverage. Your comparison should focus on (*i*) the effectiveness with respect to the probability to expose failures, (*ii*) the cost with respect to 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
Describe the three concepts software fault, software error and a software failure. Make sure that your descriptions distinguish between the three.

Question 9
During the lectures we have talked about the RIPR model, which stands for reach, infect, propagate and reveal. Describe the model and for each of the four steps explain what is required for a test execution to lead to a failure.

Question 10
One of the key features of JUnit is the automation of tests. How do you formulate an automated test that tests for null? Use the method *sum(int[] x)* for your example test. This method returns the sum of the values of an array containing integer numbers.

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 help testers get high-quality tests.

Question 11
In this course we have discussed several advantages of using coverage criteria. Two of the advantages concerns 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? *Number of arguments and level of detail determines the grade.*