

Vrije Universiteit, Department of Computer Science

Examination paper for **Software Testing**

28 May 2015 12:00-14:45

This is a closed book written exam.

No printed material or electronic devices are admitted for use during the exam.

The answers have to be given in English.

Both homework and exam are compulsory and graded on an 1 to 10 scale.

The exam grade is calculated as $(Q1+Q2+Q3+Q4+Q5 +10)/10$.

The final grade is calculated as $0.5*homework + 0.5*exam$

A pass is given if both homework and exam components are ≥ 5.5 .

	Q1 (concepts)	Q2	Q3	Q4	Q5 (code)	Σ Qi	Maximum credits= $(\Sigma Qi+10)/10$
a)	3				3		
b)	3				5		
c)	3				3		
d)	5				5		
e)	5				5		
f)	7				3		
g)	7						
Total	33	12	9	12	24	90	10

Good luck!

Q1. Concepts [33p]

- Define the term “equivalent mutant”. Give an example. [3p]
- What is the difference between verification and validation? [3p]
- What is meant with “traceability” in software testing and why is this property important? [3p]
- Define the term “reliability” and argue on safety vs. reliability. [5p]
- Apply the first step of STAMP analysis to the Therac-25 radiation overexposure accident. Identify the accident, the hazard and the safety constraint that has been violated. Each answer should consist of just one sentence. [5p]
- Explain what symbolic execution is and give an example of how it can be used to generate test cases. [7p]
- Enumerate and shortly describe four ways to perform integration testing of embedded systems. [7p]

Testing from requirements Q2-Q4 [33p]

Q2. [12p]

Consider this requirement for a web store selling rollerball pens.
A customer can order up to 500 pens. The price per item depends on the quantity and is listed below:

Number of items	Price per item (euros)
1 to 100	10
101 to 200	9
201 to 300	8
301 to 400	7
401 to 500	6

Generate test cases using equivalence partitioning analysis and boundary value analysis for a defensive testing of this requirement. Justify your test cases specifications.

Q3. [9p]

Below you can find the specifications for a store software module that provides customer benefits.

CUSTOMER
If the customer is a new customer, offer 20% discount on next order
If the customer is a repeat customer, offer free shipping
RISK LEVEL OF GOODS
If the risk level of goods is high, then
If the customer is a new customer, check their credit record
If the customer is a repeat customer, then:
If the past orders total > £500, fine
Otherwise check their credit record

Generate test cases for this module using a decision table approach.

Q4. [12p]

This is an admission requirement the Boise State's University:

A high school graduate is admitted based on the high-school grade point average (GPA) and the ACT composite score (English, mathematics, reading, science).

The GPA score is a number, $2.0 \leq \text{GPA} \leq 4.0$ (for example 2.8)

and ACT is a number, $16 \leq \text{ACT} \leq 36$, (for example 30).

Admission is possible only if $10 * \text{GPA} + \text{ACT} \geq 56$.

Design test cases for this requirement using 1x1 domain testing.

- Draw the valid domain. [3p]
- Construct the domain test matrix. [9p]

Q5. Code based testing [24p]

This Java method implements a string search algorithm. It returns the first occurrence of string `needle` in the string `haystack`, or -1 if the `needle` is not a substring of `haystack`.

```
static int match(String haystack, String needle) {
    for (int i=0; i+needle.length() <= haystack.length(); i++)
    {
        int j= 0;
        while (j < needle.length() &&
                haystack.charAt(i+j)==needle.charAt(j))
            ++j;
        if (j==needle.length())
            return i;
    }
    return -1;
}
```

For this code snippet:

- a) Draw the control flow graph. [3p]
- b) Generate a test suite that achieves 100% statement coverage. [5p]
- c) Enhance if necessary your test cases from b) to achieve 100% decision coverage. [3p]
- d) Generate a test suite that is adequate with respect to the all-defs criterion [5p]
- e) Generate a test suite that is adequate with respect to the all-uses criterion [5p]
- f) Generate a mutant and show a test case that will kill it. [3p]