



Name:

Vnumber:

SENG 275 MID TERM EXAM 1 -O2 (10%)

Instructor: Dr. Navneet Kaur Popli, Date: 30 May 2023, Time: 12:30-1:20 PM PST

Mode: Pen-Paper, Synchronous, Timed, Closed-book, to be done in question paper

Total Marks: 44, No. of pages: 7, No. of questions: 18

Note: This is an individual activity. Copying or cheating of any kind is not allowed. Supplement with diagrams, tables, graphs, and charts wherever applicable.

Q1) Having certain terminology helps testers to explain the problems they have with a program or in their software.

Below is a small conversation. Fill in the blanks with one of the following terms: failure, fault, or error. (3M)

- **Mark:** Hey, Jane, I just observed a _____ in our software: if the user has multiple surnames, our software does not allow them to sign in.
- **Jane:** Oh, that is awful. Let me check the code so that I can find the _____.
- **Jane (a few minutes later):** Mark, I found it! It was an _____. I programmed that part, but never thought of this case. I should be paying more attention to the documentation.
- **Mark:** No worries, Jane! Thanks for fixing it!

Q2) A tester was doing a code review for a System Under Test (SUT). They came across the following code: (4M)

```
public static int oddOrPos(int[] x)
{
    //Effects: if x==null throw NullPointerException
    // else return the number of elements in x that
    //      are either odd or positive (or both)
    int count = 0;
    for (int i = 0; i < x.length; i++)
    {
        if (x[i]%2 == 1 || x[i] > 0)
        {
            count++;
        }
    }
    return count;
}

// test:  x=[-3, -2, 0, 1, 4]
//      Expected = 3
```

- Zero is an even number.
- The modulo operator (in this code) returns the remainder of the two numbers after division. $26\%15=11$, $15\%6=3$, $2\%4=2$, $-11\%5=-1$, $-11\% -5=-1$, $11\% -5=1$.
- $x=[-3, -2, 0, 1, 4]$ is a test case.

(a) Identify the fault in this piece of code.

(b) If possible, identify a test case that does not execute the fault.

(c) If possible, identify a test case that executes the fault, but does not result in an error state.

(d) Fix the fault and verify that the given test now produces the expected output.

Q3) Which of the following are valid testing principles? (1M)

I) Exhaustive testing is in general impossible.

II) Exhaustive testing should be executed for code intended to be reused.

III) Testing may guarantee that a program is correct.

IV) Testing cannot guarantee that a program is correct.

V) Defects cluster together in certain areas of the product.

a) I, IV, V

b) I, V

c) I, III

d) II, IV

Q4) Which of the following could be the possible examples of what you might test in unit testing (select all that apply). (2M)

a) Multiple classes for feature implementation.

b) A whole class.

c) A single method in a class.

d) A functional use case implementation.

Q5) Match the **testing principles/concepts** (from the list shown on the right) that **best** describes the concepts or examples shown on the left. (5M)

.....Sally is a new developer at the University of Victoria, and she notices that many of the complaints about Brightspace all seem to be about the Quiz module! Which testing principle best describes this phenomenon.

1. Pesticide Paradox
2. Exhaustive Testing is Impossible
3. Absence of errors is a fallacy
4. Defect Clustering
5. Good Weather Testing

.....Testing your program to make sure that it conforms to the user requirements and the expected user inputs and testing the positive flow is often the first goal when writing test cases before writing or designing your code. Which testing principle best describes this situation.

.....Peter is an experienced developer and believes that he can fully test Microsoft Teams so that it never crashes! Which testing principle best explains why this is **NOT** possible?

.....Some years ago, Google developed a communication tool called "Google Wave" that Googlers loved, and it worked flawlessly. However, most people did not find it useful for their needs. Which testing principle best describes this problem?

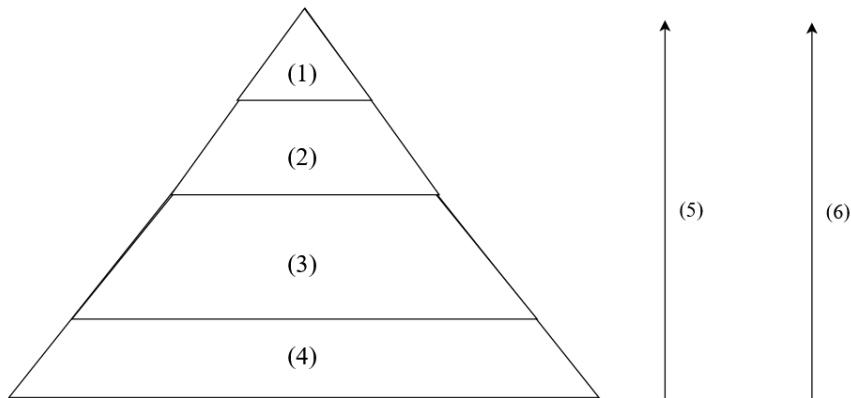
.....Jean believes that unit testing is the most effective way to use automated testing, so they put all their energy into that type of testing. Which testing principle captures that this is not the preferred approach!

Q6) What do you understand by scaffolding? (2M)

A6)

Q7) System testing is a type of white box-testing. (True/False) (1M)

Q8) Here is a skeleton for the testing pyramid. Fill in the correct corresponding terms. (6M)



Q9) A newly developed product started off with some basic unit tests but later decided to add only integration and system tests for the new code that was written. This was because a user interacts with the system as a whole and therefore these types of tests were considered more valuable. Therefore, unit tests became less prevalent, while integration and system tests became a more crucial part of the test suite. Which of the following describes this transition? (1M)

1. Transitioning from a testing pyramid to an ice-cream cone anti-pattern
2. Transitioning from an ice-cream cone anti-pattern to a testing pyramid
3. Transitioning from an ice-cream cone pattern to a testing pyramid

Q10) A program called “Software-Testing” does the following:

Given an integer ‘n’, return the string formed from the number n (using the logic explained below) followed by “@”.

- If the number is divisible by 2, use “Software” instead of the number; and
- if the number is divisible by 7, use “Testing” instead of the number, and
- if the number is divisible by both 2 and 7, use “Software-Testing” instead of the number.
- if the number is neither divisible by 2 or 7, use “n@” instead of the number ‘n’.

Examples:

- The integer 22 yields “Software@”
- The integer 77 yields “Testing@”
- The integer 42 yields “Software-Testing@”
- The integer 201 yields “201@”

A novice tester is trying to devise as many tests as possible for the ‘Software-Testing’ program and comes up with the following:

- T1 = 10
- T2 = 28
- T3 = 14
- T4 = 99
- T5 = 12
- T6=35
- T7=21
- T8=23

Which of these tests can be removed while maintaining a good test suite and why? Explain in detail. (3M)

Which concept can we use to determine the test(s) that can be removed? (1M)

Q11) Choosing the level of a test involves a trade-off. After all, each test level has advantages and disadvantages. Which one of the following is the **main advantage** of a test at system level? (1M)

1. The interaction with the system is much closer to reality.
2. In a continuous integration environment, system tests provide real and quick feedback to developers.
3. Given that system tests are never flaky, they provide developers with more stable feedback.
4. A system test is written by product owners, making it less complex and closer to reality.

Q12) What is the main reason for the number of recommended system tests in the testing pyramid to be smaller than the number of unit tests? (1M)

1. Unit tests are as good as system tests.
2. System tests do not provide developers with enough quality feedback.
3. There are no good tools for system tests.
4. System tests tend to be slow and are difficult to make deterministic.

Q13) Differentiate between smoke and sanity testing. (2M)

A13)

Q14) A software organization can be sued for damages if their application crashes and causes financial harm to the user. Given below is one way the organization can help pay legal costs: (Choose one) (1M)

- a) Create careful End User Licence Agreements which stand their ground in court
- b) Take Errors and Omissions insurance
- c) Take Software Development and Testing Insurance
- d) Show due diligence during development and testing and allocate appropriate budget to it.

Q15) For someone to successfully sue a software developer for negligence in absence of a contract, they must prove these 4 things (select all 4 that apply): (4M)

- a) Duty of Care
- b) Duty to warn
- c) Standard of Care
- d) Failure to warn
- e) Damages/Injury
- f) Failure to perform
- g) Shared liability
- h) Causation

Q16) Explain using any 2 examples how you can find errors in the requirements stage. (2M)
A16)

Q17) Code reviews are an example of validation. (True/False) (1M)

Q18) What are flaky tests? Explain any one way to spot a flaky test and any one way to handle them once you have spotted them. (3M)
A18)

END