United International University
Department of Computer Science and Engineering

CSE 4495: Software Testing and Quality Assurance

Mid-term Examination : Spring 2023

Total Marks: 30 Time: 1 hour 45 minutes

Any examinee found adopting unfair means will be expelled from the trimester / program as per UIU disciplinary rules.

Answer all the questions. Numbers to the right of the questions denote their marks.

1. (a) *ZetaSoft.Inc* is a large company that builds and maintains multiple lines of B2B softwares. As a new member of the development team you find out that developers are paired in teams of two where one extensively reviews the other person's code before it is merged to the codebase. After that, the developed APIs are tested by automated tools and scripts against their documentation. What are the two different ways of verification involved in the aforementioned process? Describe their advantages and disadvantages with examples. [2]

   (b) Briefly describe the CIA characteristics of a secured system. [2]

   (c) Define the *Dependability* of a system. What attributes need to be examined in order to establish dependability. Describe the relationship between them. [2]

2. (a) You manage an online resource sharing platform where people can share e-books and others study materials. A typical upload/download takes ten minutes on average, and an interrupted upload/download must be restarted from the beginning. The number of customers engaged in a activity at any given time ranges from about 960-1200 during peak hours. On average, your system goes down (dropping all connections) about three times per week, for an average of four minutes each time. You are given the following two options to improve the availability of your system(so that less customers face interruption when they are using your services). [3]

      i. Reducing total downtime to half (two minutes to restart on average after each crash, but number of crashes remain unchanged).

      ii. Increasing the current *MTBF* by 50% (i.e. two system outage per week, but total downtime remains unchanged).

   Analyze both possibilities to prove statistically which method alleviates more users and decide which option better achieves your goal.

   (b) Imagine you are the quality assurance lead of an online shopping service. While testing your company's upcoming mobile application you were provided the following requirements for product release- availability of at least 99.8%, *ROCOF* less than 7 failures per day and *POFOD* less than 0.004 [3]
   After the testing is done you receive the following report from your testing team - "During 10 days of testing the system processed 18759 requests.

      i. 26 times the system showed an user wrong information about a searched product.

      ii. 27 times the system disconnected the user from a live session.

      iii. 12 times the whole system crashed, and servers needed to be restarted. Each restart took 2 minutes on avg."

   Now depending on this report measure the availability and *POFOD* of your system. Also decide whether your product meets the requirements given for release.

3. (a) Describe the different components of a *Test Oracle*. [1]

   (b) You are testing a high level API named *Geolocation* which receives the name of a city/location and responds with the Latitude and Longitude of the said location? Here you are looking to test two things- [2]

      i. Whether the API returns correct coordinates.

      ii. Does the API meet response time requirements(responds within 200ms)?

   What are the types of oracles you need to use when you write test cases for this API.

   (c) What is Resilience testing? Describe types of Acceptance tests and their purposes. What is the reasoning behind executing System/UI tests in far lesser number than Unit tests? [3]

4. The product comparison tool is a high-level function exposed by the API of an online shopping platform that is useful for many purposes. It is mainly intended to check the relevance of a certain product with another product in the database. For example, a product could be completely similar to another one. It might be partially similar or not at all similar. This service will determine such similarity or relevance between two products.

ComparatorTool(Product productA, Product productB) returns comparisonCode;

A **Product** is a data structure consisting of:

- A unique identifying product code (String, four characters).
- The product vendor (String, four character short code).
- The category that the product belongs to (String, e.g. Category - Electronics).
- The sub-category in the category (String, e.g. Category - Electronics , Sub-Category - Smartphones).

comparisonCode is an integer with value:

- 1 for *Perfect match*. That means both the products are made by the same vendor and belong to same category and sub-category.
- 2 for *Similarity match*. This means the products are from different vendors but belong to same category and sub-category.
- 3 for *Partial Similarity match*. This means the products are from different vendors and also belong to different sub-categories, but their category is still the same.
- 4 for *No match*.

Now, you have to design system level test cases using the category-partition method for the ComparatorTool function. To do this, you are provided the following choices (aspects that you control and that can vary the outcome) for the two input products and the database.

- Parameter: **productA**; choices- Product Code, Vendor, Category, Sub-Category
- Parameter: **productB**; choices- Product Code, Vendor, Category, Sub-Category
- Parameter: **Database**; choices- Number of Products, number of Categories, number of Sub-categories.

(a) For each choice, identify a set of representative values(choices can be merged for better cohesion). [2]

(b) Apply at least 5(five) **ERROR**, 1(one) **SINGLE** and 2(two) **IF** constraints to your representative values derived in Q.4(a). [2]

(c) Calculate the total number of test specifications generated before and after applying these constraints. [2]

5. (a) Here are some of the configurations that can be controlled in a certain web browser: [4]

Allow Cookies	Warn About Add-Ons	Warn About Attack Sites	Warn About Forgeries
Allow	Yes	Yes	Yes
Restrict	No	No	No
Block			

Figure 1: Configurations for Question 5(a)

What will be the total number of test specifications in this case? Using Combinatorial interaction technique create a covering set that covers all 2-way combinations of these browser configurations.

(b) If you decided to perform 3-way combinatorial testing in Q.5(a), what will be the size of the covering set Also mention one such combination of settings which will be covered by this covering set that wasn't covered by your solution of Q.5(a) [1+1=2]