



United International University

Department of Computer Science and Engineering

CSE 4495: Software Testing and Quality Assurance

Mid-term Examination : Summer 2022

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) What are the properties that must be established in a system to deem it as 'Dependable'? Describe the relationship between these properties with necessary illustrations [3]
- (b) Consider the following statements- [3]
- A system can be correct yet unsafe.
 - A system can be safe but not robust.
 - A user needs to download a very large file over a slow modem, then he/she should be more concerned about the system 'availability' rather than the 'Mean time between failures(MTBF)'
- Now either support or oppose each of these statements and explain the logical reasoning behind your stand.
- (c) Imagine you are the lead developer of *FreeSpace, Inc* game studios. Your company wants to release a new mobile shooting game that will rival popular games like Free Fire, PUBGm etc. To achieve this your system needs to fulfil the following requirements-availability of at least 99.6%, a probability of failure on demand of less than 0.05, and a rate of fault occurrence of less than 4 failures per 36 hour work period. [4]
- After the testing is done you receive the following report from the testing team - "During 10 days of testing the system processed 18972 requests. Some of these requests ended in failure. Three types of failures were observed -
- 26 times the system showed an user wrong information about enemy position.
 - 27 times the game disconnected the user from a match.
 - 32 times the whole system crashed, and servers needed to be restarted. Each restart took 5 minutes on avg."

Now depending on this report measure the availability, POFOD and ROCOF of your system. Also decide whether your software is ready for release.

- (a) Show the V-model of development and elaborate on the different testing stages this model incorporates. [2]
- (b) The delivery route connection check is a high-level function exposed by the API of a International parcel shipping system. It is intended to check the validity of a single connection between two shipments in an itinerary. For example, a package may need to be delivered from Dhaka to New York, but there is a connection through London. Therefore, their itinerary is *Dhaka → London(ShipmentA)* and *London → NewYork(ShipmentB)*. This service will ensure that the connection through London is a valid one. For example, if the arrival warehouse of Shipment A differs from the departure Warehouse of Shipment B, the connection is invalid. That is, if we pass in two Shipments, and Shipment A arrives in London, but Shipment B departs from Glasgow, it is not a valid connection. Likewise, if the departure time of Shipment B is too close to the arrival time of Shipment A, the connection is invalid. If Shipment A arrives in London at 8:00, and Shipment B departs at 8:05, there is not sufficient time to complete the unloading process of Shipment A and loading process of Shipment B. [8]
- `validConnection(Shipment shipmentA, Shipment shipmentB) returns ValidityCode;`

A Shipment is a data structure consisting of:

- A unique identifying shipping code (string, four characters followed by three numbers).
- The originating warehouse code (four character string).
- The scheduled departure time from the originating warehouse (in universal time).
- The destination warehouse code (three character string).
- The scheduled arrival time at the destination warehouse (in universal time).

There is also a **Shipment database**, where each record contains:

- Four-letter Warehouse code (four character string).
- Warehouse country (three character string).
- Warehouse city (three character string).

- Minimum processing times (integer, minimum number of minutes that must be allowed for Shipment connections).

ValidityCode is an integer with value:

- 0 for OK.
- 1 for invalid warehouse code.
- 2 for a connection that is too short.
- 3 for shipments that do not connect (arriving shipment is not stored in the same warehouse as departing shipment).
- 4 for any other errors (malformed input or any other unexpected errors).

Design system test cases using the category-partition method for the validConnection function.

- Identify choices (aspects that you control and that can vary the outcome) for the two input shipments and the database.
- For each choice, identify a set of representative values.
- Apply **ERROR**, **SINGLE** and **IF** constraints. Explain the logic behind imposing these constraints

(a) Here are some of the configurations that can be controlled in a certain web browser:

[6]

Allow Content to Load	Notify About Pop-Ups	Allow Cookies	Warn About Add-Ons
Allow	Yes	Allow	Yes
Restrict	No	Restrict	No
Block		Block	

Figure 1: Configurations for Question 3(a)

What will be the full number of test specifications in this case? Using Combinatorial interaction technique create a covering array covering all pairwise combinations of these browser configurations.

- How can one use Category-partition and Combinatorial interaction techniques together? Explain with necessary examples. [2]
- What are Unit test cases? Which test cases among Unit tests and system test are executed greater in number? Why? [2]