COEN 448 Software Testing and Validation Assignment 1 – Input Domain Modeling and Unit Testing

In this assignment, we aim to practise the input domain partition, input domain modeling and coverage strategies to produce unit test cases.

The source code attached is a data structure that implement an abstract data type ADTQueue. It has three different implementation Array-based Queue (AQueue), Link-based Queue (LQueue) and Double Link-based Queue (DQueue). Now please review and inspect the code of the enqueue() and dequeue() function in the ADTQueue which is implemented by each of the class AQueue, LQueue, DQueue.

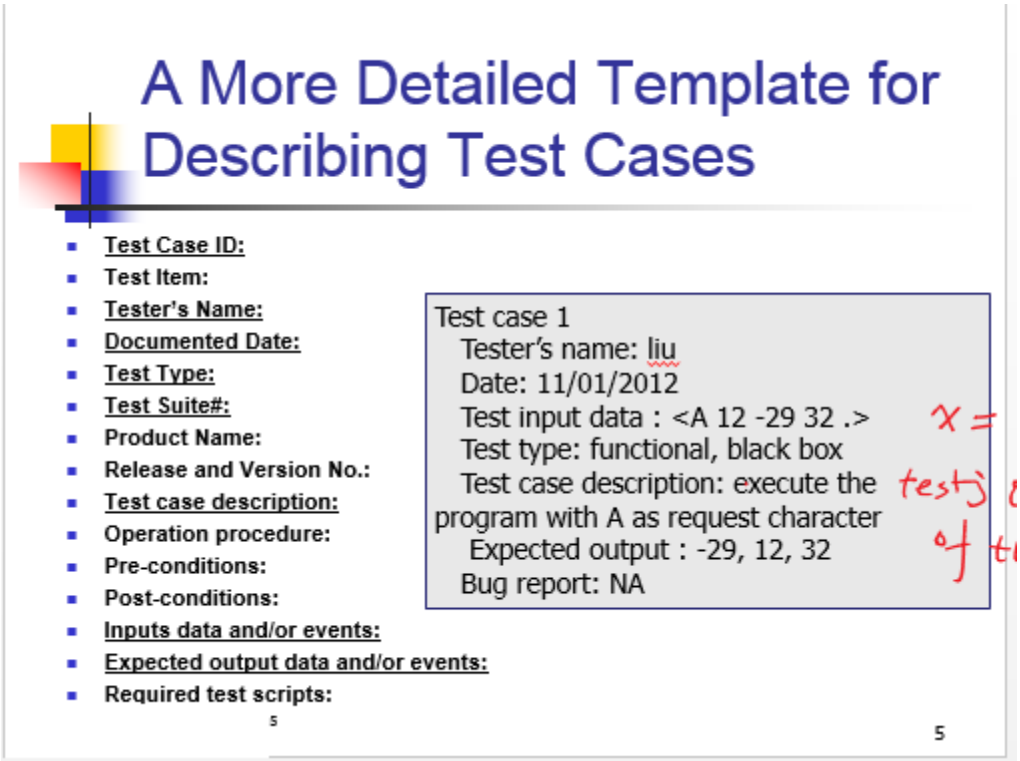
The tasks are listed below.

1) Apply the Input Domain Modeling techniques to fill in the table of program characteristics and block values for the method enqueue() and dequeue() of any class that implements ADTQueue. Please include both interface and functionality based IDM for defining characteristics.

Hint: define the table for each implementation classes, AQueue, LQueue and DQueue, and discuss in the report if they share characteristics.

	b1	…	bn
Characteristics (Interface-based IDM)			
Characteristics (Functionality-based IDM)			

2) Apply Each Choice Coverage (ECC) and list the generated combination in a table.
3) Apply Basic Choice Coverage (BCC) and list the generated combination in a table.
4) Program the code of the unit test for the combination by revising the QueueJUnitTest.java and run the unit test code to show the test state. The unit test case should have standard documentation in the code comment section. The test case description should follow the format as the lecture slides.



Deliverables (create a single archive in .zip or .tar or .gz; **NO .rar file is accepted)**

- A report to document task 1), 2), 3) and the screenshot the running the unit test from the IDE.
- Source code of the QueueJUnitTest.java