Midterm Examination Makeup

Version 2.0

Due December 16, 2017

If you complete both parts carefully and correctly, your midterm exam grade will be raised to 85, but no higher.

Part A. – Critically analyze your Midterm submission, question by question, discussing all errors of commission and omission you find. Please do not feedback my solutions to each problem. Instead, discuss problems with your submission. I would expect you to use my solutions to help you do that, but you are analyzing your errors, not my solutions.

Part B. – Provide solutions for each of the requests, below:

1. Write an Executive Summary for your Project #1 OCD, e.g., summarize the concept, structure, and critical issues you expect to discuss in the body of the report.
2. Describe in concrete detail how the CsMessagePassing communication system provided in the Project4HelpF2017 folder works. You will need to discuss the interfaces, classes, and important methods.
3. Create a list of at least seven messages needed by Project#4 communication between Clients, Repository, Build Server, and Test Harness. Please list the name of the message, its destination, source, and contents.
4. Describe the XML parsing used for the BuildRequest and TestRequest in the Clients, Repository, BuildServer, and TestHarness servers. The structure of these Requests should match the operation needs of each server.
5. Write the code for an event logger that stores log items in an in-memory stream and, on command, persists its contents to an XML file. You may choose any processing event from Project #4 as a model for the data being logged. Can you make the logger work for any Project #4 event?
6. Convert your logging facility to accept log messages asynchronously.
7. Draw a class diagram for file caching on the Project #4 Build and TestHarness servers. Explain the responsibilities of each class. Explain why this makes sense only if files and libraries are versioned.
8. Enumerate issues associated with the logging facility of B.5 and suggest solutions.
9. Where would you expect to use C# tasks in an implementation of a complete Code Repository Data Management Service.
10. Write code to list all the Repository packages depending on a specific named package using reflection.