Nova Southeastern University

College of Engineering and Computing

Fall 2019 - Master Level Course

CISC 680 - Software Engineering - CRN – 21741

Term Code: Fall 2019 (202020) Course

Dates: 08/19/2019 - 12/08/2019 – On Line

Your document for this assignment will be submitted to Canvas (Final) in one of the following file formats: ASCII, MS Word, or PDF

Advice: Remember to clearly answer each question, you must provide and intelligent understanding of the question to get complete credit, therefore concise is not a requirement, but a complete answer that addresses the complete question is.

Final Due – End of term

Question 1:

This is a software Testability Checklist

1. Operability – the better it works the more efficiently it can be tested
2. Observability – what you see is what you test
3. Controllability – the better software can be controlled the more testing can be automated and optimized
4. Decomposability – by controlling the scope of testing, the more quickly problems can be isolated and retested intelligently
5. Simplicity – the less there is to test, the more quickly we can test
6. Stability – the fewer the changes, the fewer the disruptions to testing
7. Understandability – the more information known, the smarter the testing

Choose two and explain why they have higher merit than the others, remember to provide enough depth so that you convey that you understand the depth of the question.

Question 2:

In OO Testing,

What is the difference between surface and deep structure testing and how does it benefit the overall testing of OO designs?
When should they be used when why is the environment important?
What kind of project would require each, give three examples of each?

Question 3:

In Formal modeling and verification what is the Z Specification language?
Why is it used and what are the drawbacks of formal modeling, Give three examples of software projects (applications) that would required Formal modeling.

Question 4:

In Measurement Process Activities, the following list applies:

1. *Formulation* – derivation of software measures and metrics appropriate for software representation being considered
2. *Collection* – mechanism used to accumulate the date used to derive the software metrics
3. *Analysis* – computation of metrics
4. *Interpretation* – evaluation of metrics that results in gaining insight into quality of the work product

Of the above four items, list the order of focus,, highest to lower which items requires more work than others, Give a detailed explanation of why you think this is so.

Question 5:

What are the major differences between Process and Project Metrics and why are they important to software engineering?

Question 6:

How does Software Re-engineering fit into the agile development cycle, explain in detail, how to combine re-engineering process model and the agile process model

Question 7:

If you are in an agile development cycle how would you incorporate the use of UML, does it have a place and how strongly should it be implemented?

Question 8:

Regarding Clear-Box, State-Box and Black-Box quality management approaches,
Explain the uses of each?
Give an example of each?
Why you would use 1 over the other 2 in each case?