Sample Exam

ISTQB Advanced Test Analyst

Answer Rationale

Prepared By

ASTQB

Released March 2016

TTA-1.3.1   (K2) Summarize the generic risk factors that the Technical Test Analyst typically needs to consider

#1 (1 point)
A is correct, 1, 2 and 5 are the TTA's responsibility:
      1. Should be considered by TTA per the syllabus
      2. Should be considered by TTA per the syllabus
      3. Is a responsibility of TA per the TA syllabus
      4. Is a responsibility of TA per the TA syllabus
      5. Should be considered by TTA per the syllabus
      6. A TM consideration, not covered in TTA syllabus

TTA-1.x.1   (K2) Summarize the activities of the Technical Test Analyst within a risk-based approach for planning and executing testing

#2 (1 point)
B is correct. Test results may uncover new or under-prioritzed areas that need more extensive testing. It may also determine that some areas are actually less risky than expected. A is not a TTA function. C is the responsibility of the TA. D is incorrect because testing should adjust to newly found risk areas and areas where the risk level has changed from the original risk analysis (better or worse).

TTA-2.2.1   (K2) Understand how to achieve condition coverage and why it may be less rigorous testing than decision coverage

#3 (1 point)
A is correct. Condition coverage tests all the atomic conditions, but does not necessarily test the decision outcomes. It might, but it doesn't have to. B and C are not correct because A is correct. D is not correct because the technique works with any number of atomic conditions.

TTA-2.3.1   (K3) Write test cases by applying the Decision Condition testing test design technique to achieve a defined level of coverage

#4 (2 points)
D is correct:

Test	HEAT	SMOKE	BUTTON	ALARM
2	T	T	T	T
5	F	F	F	F

Note that in test case 5, the value for SMOKE is actually a "don't care". A is incorrect because it fails to evaluate BUTTON as both true and false. B is incorrect because while inputs are all evaluated as both true and false, result is not. C is incorrect because while it achieves coverage, it is not in the minimum number of tests.

TTA-2.4.1 (K3) Write test cases by applying the Modified Condition/Decision Coverage (MC/DC) testing test design technique to achieve a defined level of coverage

#5 (2 points)
C is correct:

Test	PROXIMITY	CLOSURE	PEDESTRIAN	BRAKE
2	TRUE	TRUE	FALSE – N/A	TRUE
3	TRUE	FALSE	FALSE	FALSE
4	FALSE	TRUE – N/A	TRUE	TRUE

Each condition is tested as both true and false, the result is both true and false, plus:
    Changing PROXIMITY to false in test 2 changes result to false
    Changing CLOSURE to true in test 3 changes result to true
    Changing PREDESTRIAN to false in test 4 changes result to false

Also note that test 5 is not needed because of short-circuiting. The False in the PROMIXITY field will cause the True in the CLOSURE to not be evaluated. A is incorrect because there is no case where PROXIMITY and CLOSURE are tested individually. B is incorrect because there is no case in which changing PEDESTRIAN to false changes decision. D is incorrect because there is no case in which changing PEDESTRIAN to false changes decision.

TTA-2.5.1 (K3) Write test cases by applying the Multiple Condition testing test design technique to achieve a defined level of coverage

#6 (2 points)
D is correct. There are three conditions and the formula is 2^n where n is the number of conditions. 2x2x2 = 8. If the conditions are built into a table it would look like this:

	Age 3-6 months	No rabies shot	Healthy	A and B and C
Test 1	True	True	True	True
Test 2	True	True	False	False
Test 3	True	False	True	False
Test 4	True	False	False	False
Test 5	False	True	True	False
Test 6	False	False	True	False
Test 7	False	True	False	False
Test 8	False	False	False	False

TTA-2.6.1 (K3) Write test cases by applying the Path testing test design technique

#7 (2 points)
C is correct. One down the right side, one down the left side and one with the loop. Beizer's guidance is that only one decision outcome can be changed at a time.

TTA-2.7.1   (K2) Understand the applicability of API testing and the kinds of defects it finds

#8 (1 point)
D is correct. With this number of interfaces to test, combinatorial testing should be used to limit the APIs that you need to test. A is not correct because there are no boundaries to test in this case. You could organize the various interfaces into partitions, but this would not be the most appropriate approach because coverage will be better with combinatorial. B is not correct because condition coverage is not applicable to this scenario. C is not correct because, while you could do path testing, there is no mention of data flow testing in the question.

#9 (1 point)
B is correct. API testing is needed to be sure the services are all talking with each other and that error recovery is handled properly. A is not correct because, although you should do security testing, that will not help with the issues with retry and downtime logic. C is not correct because this is not a performance issue. D is not correct because usability has little or nothing to do with this, except maybe to notify the user that the service is down.

TTA-2.8.1   (K4) Select an appropriate structure-based technique according to a given project situation

#10 (3 points)
A is correct. This is a SIL 4 application and the specification recommends MC/DC coverage.
B is incorrect because DO-178B concerns flight systems, and the recommendation is for a non-critical system. C is incorrect because while the recommendations come from IEC-61508, they are for a less critical system. D is incorrect because DO-178B concerns flight systems, yet those recommendations are for a less critical system under IEC-61508.
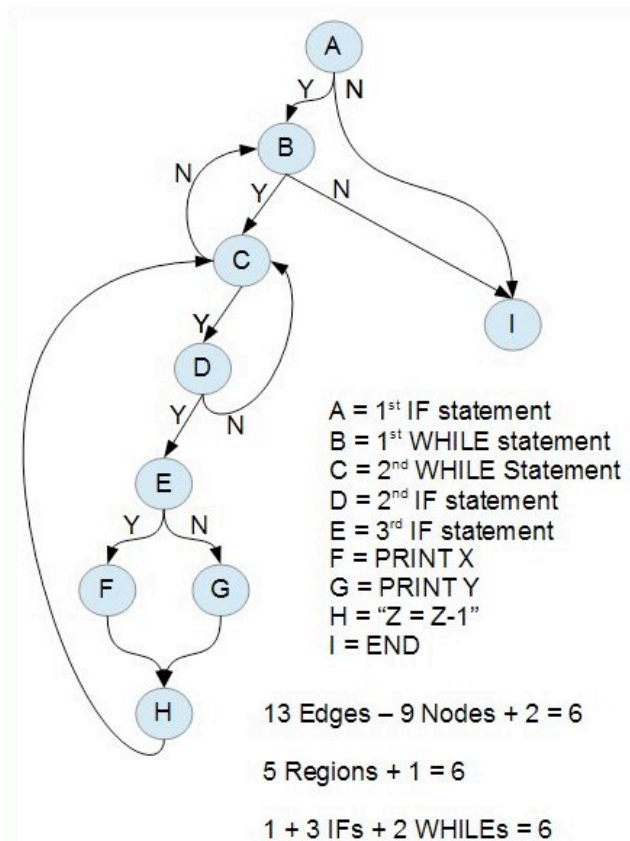
#11 (3 points)
C is correct. According to the specification, decision coverage is required.

TTA-3.2.1   (K3) Use control flow analysis to detect if code has any control flow anomalies

#12 (3 points)
B is correct. The cyclomatic complexity is 6. See the drawing below and the application of the cyclomatic complexity formulas. There are 6 basis paths:
   1.   A-I
   2.   A-B-I
   3.   A-B-C-B-I
   4.   A-B-C-D-C-B-I
   5.   A-B-C-D-E-F-H-C-B-I
   6.   A-B-C-D-E-G-H-C-B-I

A = 1st IF statement
B = 1st WHILE statement
C = 2nd WHILE Statement
D = 2nd IF statement
E = 3rd IF statement
F = PRINT X
G = PRINT Y
H = "Z = Z-1"
I = END

13 Edges – 9 Nodes + 2 = 6

5 Regions + 1 = 6

1 + 3 IFs + 2 WHILEs = 6

#13 (2 points)
D is correct. There is an infinite loop because X is never decremented and there is unreachable code because you can never get to return (22) because Net-Worth cannot be both > 10 and < 9. A is incorrect because the program will never return 22, but will always return Y which is either undefined or 201 depending on the path taken. B is incorrect because there is no reason why variable Z must be used in a control decision. C is incorrect because that is not a control path problem and is a valid computation since both are integers.

#14 (2 points)
C is correct. There is no point in having a decision if it can only ever have one outcome. A is not correct because it appears that you can exit the loop via the decision outcome. B is not correct because the complexity rating (nodes – edges + 2) is 2 (because the decision should just be inline code). D is not correct because there is nothing to indicate the operations are not in a logical sequence.

TTA-3.2.2 (K3) Use data flow analysis to detect if code has any data flow anomalies

#15 (2 points)
B is correct. These are both likely to cause the program to abort if executed. A is not correct because 2 is not a valid lifecycle. C is not correct because these are maintainability issues, but will not cause the program to fail. D is not correct because 4 is also valid.

TTA-3.2.3   (K3) Propose ways to improve the maintainability of code by applying static analysis

#16 (2 points)
C is correct. This is a maintainability issue and a static analysis tool should be used to identify the primary maintainability issues. That information can then be used to fix/update the software to improve maintainability. A might be a consideration for a short term, particularly to improve the maintainability, but would have to be done with senior supervision. B would take a long time and the problem doesn't seem to be so much with defects in the code, but rather with breaking it when changes are made. D is not correct because this is more of a description of a debugger and would only address known issues.

TTA-3.2.4   (K2) Explain the use of call graphs for establishing integration testing strategies

#17 (1 point)
A is correct. The edges represent the communication between the nodes or program units.

TTA-3.3.1   (K3) Specify goals to be achieved by the use of dynamic analysis

#18 (2 points)
C is correct. The symptoms indicate that a periodic memory corruption problem is happening which may be due to wild pointers. The crash may be occurring because memory is being overwritten when it shouldn't be. Or the wrong memory location is being accessed. Or something equally bad with memory handling. A is not correct because this isn't a performance problem. B is not correct because we're not seeking compliance to coding standards or security vulnerabilities. D is not correct because there is nothing to indicate that this problem is caused by code complexity. Re-writing the code might fix it, but it might just break other things too.

TTA-4.2.1 (K4) For a particular project and system under test, analyze the non-functional requirements and write the respective sections of the test plan

#19 (3 points)
B is correct. A potentially wide range of stakeholders should be surveyed to understand the documented and undocumented non-functional requirements. A is not correct because this is a new design. It may be useful to know of the existing issues, but that is not the best approach. C is not correct because the BA may not know about undocumented requirements or expectations. You would hope they would, but they may not have enough technical knowledge to identify these areas. D is not correct because, although it should be done, it is not enough.

#20 (3 points)
D is correct. Before working on regulation-level testing, you must first ensure that the results will be acceptable to the regulatory agency, in this case the FDA. If the results will not be accepted, there's no reason to conduct the tests. A would only be needed if this were performance testing. B is not reasonable because we already know we don't have enough devices to do this type of testing – that's why we have to use the simulator. C is not correct because we already have test automation code that works on the simulator and we know we can use that code. The specifications for the simulator might be nice, but aren't really a requirement at this point.

TTA-4.3.1　(K3) Define the approach and design high-level test cases for security testing

#21 (2 points)
B is correct.
1. Security testing has to occur at all levels, not just primarily at the system test level
2. Correct.　All levels should be covered in security testing
3. Correct.　Testing in production is necessary because of the changing risks
4. Correct.　Static testing is needed and there are good security static analysis tools available
5. Resource utilization benchmarking doesn't really make sense because memory and disk usage would not be expected to change dramatically as a result of security changes.
6. Correct.　Dynamic security testing is needed.　This includes such things as penetration testing.
7. While this would be nice, maintainability goals do not have to be met prior to starting security testing
8. Correct.　Performance testing is needed because security changes may significantly affect the performance of the software.
9. Automating security tests is not easy to do and may not be an efficient use of resources.　Planning the automation before any testing is done is not an efficient approach.　Better to do the testing manually and then decide what could/should be automated.

TTA-4.4.1　(K3) Define the approach and design high-level test cases for the reliability quality characteristic and its corresponding ISO 9126 sub-characteristics

#22 (2 points)
A is correct.　Maturity tests should be conducted and measurements for MTBF and MTTR should be gathered and reviewed.　B is not correct because failover looks at changing to a separate set of hardware to recover.　This is usually done after a catastrophic hardware/network/power failure, not a data problem.　C is not correct because the goal is to keep the system up.　MTTR will touch on the recoverability time, but not data quality.　D is not correct because the concern is not around handling stress but rather handling normal loads and daily occurrences.　That said, on a system this critical, clearly all testing types should be conducted.

TTA-4.5.1　(K3) Define the approach and design high-level operational profiles for performance testing

#23 (2 points)
A is correct.　This is an operation profile that describes what a typical user would do to interact with the system.　This profile can be used as the basis for performance tests.　B is looking at resource utilization.　C is looking at maintainability.　D is looking at performance, although the requirement is not very clear… what is "standard"?

TTA-4.x.1　(K2) Understand and explain the reasons for including maintainability, portability and resource utilization tests in a testing strategy and/or test approach

#24 (1 point)
D is correct.　Maintainability testing will help ensure that the maintainability objectives are achieved.　When maintainability is good, the cost of ownership is reduced because maintenance is faster and cheaper.　B is not correct because this is a used to determine performance and resource utilization.　C is not correct because this will not improve the long-term cost of ownership – although automating the regression tests could help reduce testing time.　A is not correct because this is done when the product is changed in production. Without maintainability, maintenance testing will take longer.

TTA-4.x.2 (K3) Given a particular product risk, define the particular non-functional test type(s) which are most appropriate

#25 (2 points)
D is correct. Scalability testing is needed for this scenario and scalability testing is a part of performance testing. A is incorrect because time performance is the concern here. B is incorrect because time performance is the concern here and nothing in the scenario indicates a load redistribution capability. C is incorrect because the systems should be built to handle the expected load. Planning to update after preventable failures is not a best practice.

#26 (2 points)
D is correct. You need to verify memory usage of the software and this is done through testing the resource utilization, in this case the memory. A, B and C are all forms of performance testing and may be done in combination with the resource utilization, but if the resource utilization is not within limits, the others don't matter. This is particularly important on a test device which might have a different memory configuration than the real device. See section 4.6.

TTA-4.x.3 (K2) Understand and explain the stages in an application's lifecycle where non-functional tests should be applied

#27 (1 point)
C is correct. Maintainability is particularly important when the software is expected to change after it has been released to production. A is not correct because usability is not a factor in determining the need for maintainability. B is not correct because although portability can affect the overall usage and long term acceptance of a product, it is not a critical indicator for maintainability testing. D is not correct because this is dealing with reliability rather than maintainability. See 4.7.

#28 (1 point)
B is correct. End to end transactions are usually used for performance testing and that requires most, if not all, the functionality to be implemented and working. A is not correct because security testing should be conducted throughout the development and then again on the finished product. C is not correct because this can be done with static testing as soon as there is code. D is not correct because each element can be tested for adaptability prior to full integration. See 4.5.3.

TTA-4.x.4 (K3) For a given scenario, define the types of defects you would expect to find by using non-functional testing types

#29 (2 points)
C is correct. You know, if nothing else, that you have a usability problem because you can't see the messages. The cause of the messages will still have to be investigated. They could be because of a reliability problem, a functional problem, a performance problem…. The only thing you know for sure is that there is a usability issue.

#30 (2 points)

C is correct. Testing uninstall is an important part of installation testing. A refers to performance testing. B refers to security testing. D refers to maintainability testing. See 4.8.1.

TTA 5.1.1   (K2) Explain why review preparation is important for the Technical Test Analyst

#31 (1 point)
B is correct. For a technical review, the TTA should be verifying what will be tested when the software is implemented. A is the job of a TA. C is usually the job of a technical writer, but may become the job of the TTA but this will occur after the review. D should occur as part of planning, not part of the review.

TTA 5.2.1   (K4) Analyze an architectural design and identify problems according to a checklist provided in the syllabus

#32. (3 points)
A is correct per the syllabus. Load balancing and replication of data are both areas for concern particularly in an environment this complex and with a high amount of data. B is not correct because caching of transactions is not indicated in the question. C is not correct. Anti-patterns should not result from defensive programming and defensive programming is not a requirement for failover systems (although probably a good idea). D is not correct. There is no indication of process isolation concerns (outside of the Yankbots) and nothing to indicate a "lazy" database design. E is incorrect. This is an excellent choice, but the developers have assured us that the system is safe from the Yankbots. And we trust them!

#33. (3 points)
C is correct. Given this information, it appears that the Online Transactional Processing (OLTP) that the users are using has not been isolated from the Online Analytical Processing (OLAP) that the reports are using. A is not correct because this is not an issue with sharing the processors correctly. B is not correct because lazy instantiation has to do with starting up classes only as needed. D is not correct because transaction concurrency deals with processing multiple transactions at the same time. The reports are sets of multiple transactions.

TTA 5.2.2   (K4) Analyze a section of code or pseudo-code and identify problems according to a checklist provided in the syllabus

#34. (3 points)
D is the correct answer. 2 and 4 are correct.

1 is not correct because the code is well and consistently formatted. It would benefit from comments though.
2 is correct. If n = 0, then you will be dividing by zero for the average=sum/n calculation.
3 is not correct because the loops all terminate.
4 is correct. The storage is not efficient. A large array of floats is created but never used. Floats take up more space than integers or real numbers.

#35. (3 points)
D is correct.

1. Correct. The variables do not have useful names (x) and the variable "password" is not defined.
2. Correct. Variables y and z are not used
3. Incorrect. There are no divisors in this code
4. Correct. Loop counters are not initialized – x is not given a value
5. Incorrect. There are no endless loops, assuming x gets defined
6. Correct. The statement noting that you exceeded the number of tries should be outside the loop
7. Correct. Magic number constant "3" is used for the number of times a password is checked.

TTA-6.1.1 (K2) Explain technical aspects to consider when multiple tools are used together

#36. (1 point)
C is correct. This is a big concern when automating the defect logging from a static analysis tool because so many warnings are generated. The TTA should be able to limit the automatic defect logging to just those items classified as "errors" rather than automatically logging all the warnings as well. A and B are concerns, but do not matter unless C can be resolved. D should not be an issue because the developers should already be familiar with the defect management tools.

TTA-6.2.1 (K2) Summarize the activities that the Technical Test Analyst performs when setting up a test automation project

#37 (1 point)
B is correct. This is usually the most important interface because these tools can be used to schedule test execution, record results and track defects. A is not correct because there is usually no need to interface directly with the CM tools used for the target source code. C might be helpful but isn't the primary interface. Continuous deployment would be more likely. D is not correct because the test automation tools would not normally interface with either of these.

TTA-6.2.2 (K2) Summarize the differences between data-driven and keyword-driven automation

#38. (1 point)
B is correct. The external source may be a spreadsheet or a database or some similar form that the script can access to obtain the needed data. A is not correct because keyword-driven tends to be more maintainable than data-driven. C is not correct because the data is external to the script. D is not correct because action words are included in keyword-driven input sources, not data-driven. Also, for keyword-driven, the expected result should also be supplied.

TTA-6.2.3 (K2) Summarize common technical issues that cause automation projects to fail to achieve the planned return on investment

#39. (1 point)
D is correct. Handling failures has to be tuned to the actual failure. A, B or C could all be correct or could be completely the wrong thing to do. See 6.2.1.

TTA-6.2.4 (K3) Create a keyword table based on a given business process

#40. (2 points)

D is correct. This includes everything that the script should need to conduct the tests. A is not correct because it doesn't provide the inputs or the expected results. B is not correct because it doesn't provide the message text that is to be verified by the script. C is not correct because it contains multiple keywords. There should be only one keyword. The test script might use multiple tables to drive to the point where it can exercise the requested test.

TTA-6.3.1 (K2) Summarize the purpose of tools for fault seeding and fault injection

#41. (1 point)
C is correct. Fault injection is used to test how the software will handle unexpected and incorrect inputs. A is not correct because fault seeding is used to evaluate the ability of the testware to detect defects in the code. B and D are made up.

TTA-6.3.2 (K2) Summarize the main characteristics and implementation issues for performance testing and monitoring tools

#42. (1 point)
A is correct. The most likely problem is that the scaled down system will not have the hardware and network bandwidth to simulate a production-like load. B is not correct because a scaled down system can still have accurate production data, maybe not all of it, but the quality should be as good as production. C is not correct because you can generate a huge number of users depending on licensing constraints. You may not be able to get them all working though! D is not correct because the operational profiles should still be accurate even if you can't use all of them in the real test.

TTA-6.3.3 (K2) Explain the general purpose of tools used for web-based testing

#43. (1 point)
B is correct. These tools are often used for standards compliance testing, including accessibility. They may be used for the other cases as well, but that is not a common usage. A and C are not correct. While the load may be generated via an automated tool, that would not be a web-based testing tool as defined in the syllabus. D is not correct because this would be a targeted security tool rather than a generic web-based tool.

TTA-6.3.4 (K2) Explain how tools support the concept of model-based testing

#44. (1 point)
B is correct. The tools take the model that has already been built via another tool and execute it to help the tester discover the paths that should be tested. These paths can then be documented into test cases. D is not correct because the tool itself doesn't test the model so much as it is used to document the paths through the model.

TTA-6.3.5 (K2) Outline the purpose of tools used to support component testing and the build process

#45. (1 point)
A is correct. This is a capability of a xUnit framework tool. These test objects are generated to assist with component level testing which is usually conducted by the developer but may also be done by the TTA.