

Lab report

Weights We are using QUB's standard Mark Schemes And Classifications: www.qub.ac.uk/directorates/AcademicStudentAffairs/AcademicAffairs/ExaminationsandAssessment/MarkSchemesandClassifications/

100	Exceptional 1st	1st	2.1	2.2	3rd	Marginal Fail	Fail
	Exceptional and exemplary work : •Has carefully followed the lab document and module knowledge from the lectures and has gone above that level •Showing thorough and systematic understanding of module content. •Clear grasp of issues involved, with evidence of innovative use and original use of learning resources. •Knowledge beyond module content. •Clear evidence of independence of thought and originality. •High critical judgement and confident grasp of complex issues.	Outstanding answer: •Has carefully followed the lab document and module knowledge from the lectures. •Showing thorough and •Methodological rigour. •Originality. •Critical judgement. •Use of additional learning resources.	Very good, comprehensive answer showing: •Has met the requirements of the lab document and module knowledge from the lectures. •Very good knowledge and understanding of module content. •Well-argued answer. •Some evidence of originality and critical judgement. •Sound methodology. •Critical judgement and some grasp of complex issues.	Good answer showing: •Good knowledge and understanding of the module content. •Reasonably well argued. •Largely descriptive or narrative in focus. •Methodological application is not consistent or thorough.	Adequate answer: •Lacking methodological application. •Adequately argued. •Basic understanding and knowledge. •Gaps or inaccuracies but not damaging.	Failing answer: •Little relevant material and/or inaccurate answer or incomplete. •Disorganised. •Largely irrelevant material and misunderstanding. •No evidence of methodology. •Minimal or no relevant material	Nil Submission; or, answer meeting none of the necessary requirements with: •No material of value to the question asked. •No recognition of the question.
	(100% – 80%)	(79% – 70%)	(69% – 60%)	(59% – 50%)	(49% – 40%)	(39% – 35%)	(34% – 0%)
2	Introduction						
7	Test plan for white-box testing						
30	Design of your white-box test-cases: Description of how you have designed additional test case (test methods) to improve code coverage for methods of claases: Range and DataUtilities						
5	Showing that the coverage threshold is achieved for each of the two classes (Range and DataUtilities)						
5	Output of test-suite execution (there should be no "errors", given by the test suite, in the JUnit output)						
5	Comparison of black-box testing and white-box testing. Usnig examples and direct experiences learned in your labs 2 and 3, and also insights from the lecture discussions						
10	Manual data-flow coverage analysis for Range, constrain () method						
10	Manual mutation analysis (testing)						
2	Discussion of Teamwork and division of work (should be detailed)						
2	Discussion of Difficulties encountered, challenges overcome, and technical lessons learned (should be detailed)						
2	Using the provided template Word file, writing quality, and general formatting of the report						

JUnit test-suite test-code

5	Using GitHub properly: -The GitHub repo ONLY includes the Java code files that the lab document has asked to be developed, and not the entire project folders -Both students should have committed to the project files -Meaningful GitHub account IDs have been chosen, including student name, to ensure that it is easy to identify who has committed what. Account IDs are not cyphered strings such as xyz_, ghost2020, etc. -We should see regular commits to the GitHub repo, during the duration of lab time period -"Collaborator" access given to lab-work graders? We need to be able to leave comments in your code
15	Quality of test-code: -Readability and understandability of test-code (Are the JUnit test methods easy to follow, through commenting or style etc?) -Have the naming conventions of test methods, variables, etc, proposed by the lab doc, been properly followed? -Exception handling has been used properly in test-code