

PGT Exam Performance Feedback

2018/2019 Semester 1

COMP60411 Modelling data on the Web

Uli Sattler
Bijan Parsia

Comments Please see the attached report.

Exam Performance Feedback COMP60411 2018/2019

The exam consisted of 25 multiple choice questions worth 25 marks, and 4 essay-style questions worth 18 marks. Of 38 students on the course unit, 37 took the exam, with an average score of 57% (STD 11.56); the average score on the MCQs is 59%, and it is 57% for the essay questions.

We ran an Item Analysis on the whole exam: we had a good spread of hard-easy questions, and reviewed the 4 questions with poor discrimination (they were ok questions with attractive distractors).

COMP60411 Exam 2018-19

Analysis Last Run 25 January 2019 16:35. Run Item Analysis again to display the latest question data

Test Summary						Discrimination	Difficulty
43	29	0	37	24.92	01 hr 48 min	17 Good Questions	9 Easy Questions
Possible Points	Possible Questions	In Progress Attempts	Completed Attempts	Average Score	Average Time	8 Fair Questions	17 Medium Questions
						4 Poor Questions	3 Hard Questions
						0 Cannot Calculate	

We comment on the 4 essay-style questions:

Question 26 was pure bookwork on the core subject of the course unit and mostly answered well (mean=3.35/6; min = 1.5; max=5.0), though a surprising number of students had difficulties relating parsing/serialising with the internal/external representation and (DOM)trees/XML documents in a suitable, clear way.

Question 27 required students to remember how a RelaxNG schema roughly looks and works like, and to create one that captures the example. This was a relatively easy question (mean=3.36 out of 4; min = 0.0; max=4.0): most students got the (very simple) regular expressions for the element content correct though some struggled with the optional elements and/or with attributes.

Question 28 was a more advanced question and required a good understanding of XSD datatypes and the application of this understanding to the running example. Performance was not great (mean=1.03 out of 3; min = 0.0; max=2.5): some students confused complex/simple types with datatypes and thus failed to answer the question; of those who understood that this was about datatypes, most spotted that `xs:integer/xs:date` (or `dateTime`) would be useful for attributes like "id" and "time"; only few noticed that it would be useful to have an enumerated datatype for the strings that can occur in "code"; even fewer noticed that, for "id" to indeed be an identifier, it has to be unique in a document, and that we can make use of a uniqueness constraint in XSD to ensure that.

Question 29 asked for the description (and thus imagination/recall) of a suitable example application and the explanation of various aspects; students found this rather difficult (mean=2.53 out of 5; min = 0.0; max=4.5). The three biggest issues were: (1) Not giving a clear, *concrete* example where the data needs clearly drive polyglot-ness (this example didn't have to hit 3 types of data; two would have been enough and a

better use of space). (2) Unconvincing data needs: saying "We can store this in tables so SQL, but that needs trees so XML" doesn't explain why it's intrinsically tabular or treelike. (3) De-normalisation does not *require* switching data models: SQL tables can be denormalised!