Revision Guide for A-Level Computer Science Unit 1.2.3 (SLR 06)

1.2.3 Software Development

Candidates need to understand the different models that can be followed to produce a program (explicitly the waterfall lifecycle, agile methodology, extreme programming, the spiral model and rapid application development).	Before Revision☆☆☆☆☆ After Revision☆☆☆☆☆
Candidates need to understand the tasks, processes, benefits and drawbacks of each model and the similarities and differences between each.	Before Revision☆☆☆☆☆ After Revision☆☆☆☆☆
Candidates need to understand where each model is most suitable to use, and be able to justify the use in a situation.	Before Revision☆☆☆☆☆ After Revision☆☆☆☆☆
Candidates need to be able to write algorithms using flow charts, pseudocode and/ or program code.	Before Revision☆☆☆☆☆ After Revision☆☆☆☆☆
Candidates need to be able to follow the code as shown in the OCR pseudocode guide, but are not expected to write code in this.	Before Revision☆☆☆☆☆ After Revision☆☆☆☆☆
Candidate's code is not expected to be syntactically correct, but must use appropriate code structures.	Before Revision☆☆☆☆☆ After Revision☆☆☆☆☆
Candidates should have experience of using black box testing, white box testing, alpha testing and beta testing whilst producing their own programs. **(covered in programming lessons and programming project)**	Before Revision☆☆☆☆☆ After Revision☆☆☆☆☆
Candidates need to understand how each testing strategy can be used in a situation, and the benefits and drawbacks of each method, and apply this to a given situation to recommend appropriate testing strategies. **(covered in programming lessons and programming project)**	Before Revision☆☆☆☆☆ After Revision☆☆☆☆☆
Candidates should have experience of using suitable test data to test their own programs. **(covered in programming lessons and programming project)**	Before Revision☆☆☆☆☆ After Revision☆☆☆☆☆
Candidates need to understand the use of test data and apply this to a given program. **(covered in programming lessons and programming project)**	Before Revision☆☆☆☆☆ After Revision☆☆☆☆☆
Candidates need to understand how dry runs can be used in the development and testing of programs, and be able to use dry runs to test given code. **(covered in programming lessons and programming project)**	Before Revision☆☆☆☆☆ After Revision☆☆☆☆☆
Candidates should understand the need for and importance of end user feedback. **(covered in programming lessons and programming project)**	Before Revision☆☆☆☆☆ After Revision☆☆☆☆☆

Slideshow (Shared area)	PG Online Textbook	Teach-ICT	CraignDave (Youtube)
Slides 1 to 33 **Read ☐** **Starter Activities ☐** **Tasks ☐**	Section 2: Chapters 11 & 12 Pages 52 to 60 **Read ☐** **Questions ☐** **Exercises P56 ☐** **Exercises P61 ☐**	1.2.3 Software Development **Theory ☐** **Lesson Tasks ☐**	OCR A'Level Development Methodologies Part 1 **Watched ☐** **Cornell Notes ☐** OCR A'Level Development Methodologies Part 2 **Watched ☐** **Cornell Notes ☐** OCR A'Level Algorithms **Watched ☐** **Cornell Notes ☐** OCR A'Level Test Strategies **Watched ☐** **Cornell Notes ☐** OCR A'Level Test Data & User Feedback **Watched ☐** **Cornell Notes ☐**