KESTEVEN AND SLEAFORD HIGH SCHOOL

Computer Science Scheme of Learning

Year 10 – Term 5

Intent – Rationale

This term focuses on the role and types of testing and the use of IDE debugger tools, before teaching students how to tackle a more substantial programming project using a waterfall lifecycle: Analyse, design, build, test and evaluate.

Sequencing – what prior learning does this topic build upon?	**Sequencing – what subsequent learning does this topic feed into?**
• Year 8 & 9 Programming Units	• Y10 Term 6: Programming Project • Y11 Examination Paper Preparation • A-Level Computer Science programming and NEA
What are the links with other subjects in the curriculum?	**What are the links to SMSC, British Values and Careers?**
• DT: Project Lifecycles	• GB4e and GB4h
What are the opportunities for developing literacy skills and developing learner confidence and enjoyment in reading?	**What are the opportunities for developing mathematical skills?**
• GCSE Computer Science NEA Programming Guide: Python Edition by Alan Milosevic (Author), Dorothy Williams (Author) *(Note: Although the NEA has been removed from J277 the course still requires that students complete a similar task/range of tasks)*	• N/A

KESTEVEN AND SLEAFORD HIGH SCHOOL

Computer Science Scheme of Learning

Year 10 – Term 5

Intent – Concepts

What knowledge will students gain and what skills will they develop as a consequence of this topic?

Know

- Understand the difference between iterative testing (testing modules of a program during development) and terminal testing (testing the program at the end of production to ensure it meets the success criteria).
- Understand the different types of error that may occur when programming (Syntax errors as errors which break the grammatical rules of the programming language and stop it from being run/translated, Logic errors as errors which produce unexpected output, ad runtime errors that cause a program to terminate unexpectedly)
- Understand when and how to choose appropriate test data (Normal test data as data which should be accepted by a program without causing errors, Boundary test data as data of the correct type which is on the very edge of being valid, Invalid test data as data of the correct data type which should be rejected by a computer system, Erroneous test data as data of the incorrect data type which should be rejected by a computer system)
- Understand the tools that an IDE provides and how each of the tools and facilities listed can be used to help a programmer develop a program, including the roll of the debugger, gaining practical experience of using a range of these tools within at least one IDE.
- Understand the stages of the waterfall project management lifecycle: Analyse, Design, Build, Test and Evaluate

Apply

- Be able to clearly describe the purpose of, and differences between, iterative and terminal testing.
- Be able to identify suitable test data for a given scenario.
- Be able to create, carry out and evidence a test plan.
- Be able to write success criteria and produce designs, including a test plan, for their chosen programming project.

Extend

- Understand the limitations of the waterfall lifecycle and the potential benefits of more modern, agile approaches.
- Understand the role of other types of testing, including usability, alpha and beta testing, especially in relation to qualitative success criteria and usability.

What subject specific language will be used and developed in this topic?		What opportunities are available for assessing the progress of students?
• Iterative testing • Terminal testing • Syntax error • Logic error • Runtime error • Boundary/extreme data • Erroneous data • Valid/invalid data • Test plan • Test evidence • Corrections • refinements	• IDE (integrated development environment) • Debugger • Watch-list • Break point • Step-into and step-over • Waterfall lifecycle • Analyse • Requirements • Success criteria • Qualitative • Quantitative • Design • Evaluate	• Class Notes and in-lesson observation • Kahoot starters/plenaries and verbal questioning • Summative: Programming project (completed in T6)

Intent – Concepts

Lesson title	Learning challenge	Higher level challenge	Suggested activities and resources
			See P drive for lesson presentations/resources