Bishop Heber High School

Subject	Year	Term
Computer Science	13	1

Topic

2.3 Algorithms // 3.2 NEA – Design of the solution // 3.3 NEA – Developing the solution

Content (Intent)

Prior Learning (Topic) GCSE equivalent content (if studied)

2.3.1 Algorithms
(a) Analysis and design of algorithms for a given situation.
(b) The suitability of different algorithms for a given task and data set, in terms of execution time and space.
(c) Measures and methods to determine the efficiency of different algorithms, Big O notation (constant, linear, polynomial, exponential and logarithmic complexity).
(d) Comparison of the complexity of algorithms.
(e) Algorithms for the main data structures, (stacks, queues, trees, linked lists, depth-first (post-order) and breadth-first traversal of trees).
(f) Standard algorithms (bubble sort, insertion sort, merge sort, quick sort, Dijkstra's shortest path algorithm, A* algorithm, binary search and linear search).

3.2 NEA - Design of the solution
3.2.1 Decompose the problem
3.2.2 Describe the solution
3.2.3 Describe the approach to testing

3.3 NEA – Developing the solution
3.3.1 Iterative development process
3.3.2 Testing to inform development

Future Learning (Topic) N/A – as per specification

What Knowledge and Skills will be taught (Implementation)	How will your understanding be assessed & recorded (Impact)
See above.	End of topic exam-style tests – approx. 45-minute duration. Low-stakes quizzing via Smart Revise and other platforms. Full practice papers for summative assessments. In class questioning via verbal and MWB

How can parents help at home?

Ask students what they are learning; Check assignments are completed on time; Encourage independent learning beyond just set directed work. Students should complete approximately 9 hours of self-directed work per fortnight.

Helpful further reading/discussion (including Reading and Vocabulary Lists)

Reading	Vocabulary Lists	
Course textbook ISBN 978-1910523056 OCR AS and A Level Computer Science, by Heathcote and Heathcote. Topic PPTs (Craig and Dave / PG Online as required) Isaac / Ada Computer Science websites	Data structures: Arrays, Records, Queues, Stacks, Trees, Graphs Algorithms: Sorting (Bubble, Merge, Insertion), Searching (Linear, Binary) Big O notation: Time and space complexity	Algorithm design: Pseudocode, Flowcharts Divide and conquer Greedy algorithms Dijkstra's algorithm A algorithm* Software development lifecycle: Waterfall, Agile, Spiral, RAD