YEAR	TOPICS	TERMLY INDEPENDENT WORK

7	1. Autumn Term
	  1.1. Basic Digital Skills
	  1.2. Components of a Computer System
	  1.3. Data Representation Techniques
	2. Spring Term
	  2.1. Introduction to programming
	  2.2. Computational thinking
	3. Summer Term
	  3.1. Further programming with text
	  3.2. Introduction to Python
		1. Autumn Term
		  1.1. Research into the history of Computer Science
		  1.2. Designing a computer system challenge
		2. Spring Term
		  1.1. Investigate Redstone Logic in Minecraft
		  2.1. Modelling exercise: Party planner
		  2.2. Python: Chilli Challenges (Turtle)
		3. Summer Term
		  3.1. Python: Chilli Challenges (Quizzes)

8	1. Autumn Term
	  1.1. Flowcharts and Control
	  1.2. Physical computing with MicroBits
	2. Spring Term
	  2.1. Further introduction to Python
	  2.2. Developing for the web
	3. Summer Term
	  3.1. Further programming
	  3.2. Excel skills
		1. Autumn Term
		  1.1. Flowol models
		  1.2. Investigation into embedded systems
		2. Spring Term
		  2.1. Python: Chilli Challenges
		  2.2. Minecraft Data Representation
		3. Summer Term
		  3.1. HCI design

9	1. Autumn Term
	   1.1. Networking – Topologies and Protocols
	   1.2. 3D modelling and animation
	2. Spring Term
	   2.1. Modular programming in Python/c#
	   2.2. Mobile apps development
	3. Summer Term
	   3.1. Physical Computing
	   3.2. Minecraft Edu
		1. Autumn Term
		   1.1. Develop revision guide on networking
		   1.2. Using blender
		2. Spring Term
		   2.1. Creating mobile apps
		3. Summer Term
		   3.1. Robotics investigations
		   3.2. Open Ended programming project

10	1. Boolean Logic 2. Data Representation 3. Designing, creating and refining algorithms 4. Machine Architecture 5. Network Topologies 6. Practical Programming	1) A wide range of extension programming tasks are available throughout the year, requiring students to develop their analytical, design and development skills within Python 2) Develop revision materials in collaborative work area 3) Investigation into Assembly Language 4) Investigation of alternative languages
11	1. Wired and Wireless networks 2. Topologies, protocols and layers 3. Defensive Design 4. Practical programming projects 5. System software 6. Language Translation 7. Ethical, legal, cultural and environmental impact	1) Develop revision materials in collaborative work area 2) Research directly relating to the programming project

PLEASE NOTE:

- This overview sets out a general summary of the basic curriculum taught. It is not an exhaustive list of what may be taught and subject teachers may follow the above in a different order. Further details may be obtained from the Head of Department, if required.
- The Independent Work indicated represents core, headline tasks per term; weekly/fortnightly independent/home work is set in all subject areas, and details are noted in Teams.