Computer Science: GCSE

# Course Overview

* **Exam Board** – OCR
* **Usual Age Range** – 14-16
* **Qualification** –1 GCSE
* **Curriculum Time** – Five 50 minute lessons per week in class plus additional work in Independent Learning Time. This time will also include Careers Advice and Guidance sessions linked to Digital roles.
* **Assessment** – this curriculum is assessed via:
  + 2 x 90 minute exams
  + Students will be provided with the opportunity to undertake a programming task, either to a specification or to solve a problem
* **Grading** – 1 to 9
* **Full specification** - <https://www.ocr.org.uk/qualifications/gcse/computer-science-j277-from-2020/specification-at-a-glance>

# Curriculum Intent

The **intent** of the Computer Science curriculum is to give UTC students an opportunity to develop their understanding of the fundamental principles and concepts of Computer Science along with practical programming skills. The intent is to ensure students have useful knowledge, understanding and skills that can be applied in any Digital Technology setting in their future career and of particular use to students considering a career in computing, games development and cyber security.

The further intent of the Curriculum is to give students useful technical skills around programming such as the ability to write an effective flow chart, produce Pseudocode, develop an understanding for basic programming skills, understand the components that make up a digital system and develop their computational thinking.

Students are supported and encouraged to develop their **love of reading** and literacy skills on this course, by reading related computing news and articles and by completing regular extended writing activities.

Students are encouraged to develop their **numeracy** on this course by applying the mathematical skills relevant to Computer Science: including number systems of binary and hexadecimal; binary additional as well as calculating file sizes for images and sound.

Suggested next step **destinations** after completion include A Level Computer Science, Level 3 Technical IT or Extended Project Qualification.

Related **careers** include working as a software developer; network support; games developer; cyber security specialist; systems analysis. This intent of the Curriculum is to also provide a good baseline knowledge, skills and understanding for students who undertake an Apprenticeship.

# Remote Learning and Revision

Students will benefit additional study of Computer Science GCSE Theory for the exam revision and also if they are absent from the UTC but well enough to complete remote learning. It is also helpful to practice Python coding regularly. Students can communicate with the teacher via Teams or via email if absent from school.

* Seneca – <https://bit.ly/2OSs4ZI>
* CGP Study Guide –
* Practice Assessments and papers - <https://www.ocr.org.uk/qualifications/gcse/computer-science-j277-from-2020/assessment/>
* Python Programming - <https://www.codecademy.com/catalog/language/python>
* Oak acdemy: <https://classroom.thenational.academy/subjects-by-key-stage/key-stage-4/subjects/computing>
* GCSE Bitesize: <https://www.bbc.co.uk/bitesize/examspecs/zmtchbk>
* Students can access all lesson materials on Teams along with a Python tutorial pack

Some other useful websites:

* Simple notes, diagrams and activities - <https://www.bbc.co.uk/bitesize/examspecs/zmtchbk>
* Video Guides mapped to topics - <https://student.craigndave.org/J277>
* Alternative web based revision sites - <https://www.computerscience.gcse.guru/>  and <https://getrevising.co.uk/resources/level/gcse/subjects/computing>
* Topic and module quizzes: <https://quizizz.com/admin>

# Curriculum Overview

The learning in Computer Science (*1 GCSE*) is sequenced as follows.

*Note: the full Curriculum Plans are available on request to* [*info@nef.tynecoast.academy*](mailto:info@nef.tynecoast.academy)

**Key Topics**

* Programming
* Fundamentals of Data Representation
* Algorithms
* Logic and Languages
* Ethical, Legal, Cultural and Environmental Impacts of Digital Technology
* Systems Architecture, Memory and Storage
* Computer Networks, Connections and Protocols
* Network Security and Systems Software

Year 10:

**Half Term 1**

* Python coding: Numbers and basic operations; values, variables and expression; data types.
* [Systems architectures](https://www.bbc.co.uk/bitesize/guides/zbfny4j/revision/1), [Memory](https://www.bbc.co.uk/bitesize/guides/zd4r97h/revision/1) and [Storage](https://www.bbc.co.uk/bitesize/guides/zd4r97h/revision/5)

**Half Term 2**

* Python coding: Functions; control functions
* [Data representation](https://www.bbc.co.uk/bitesize/guides/zfspfcw/revision/1): units of storage; binary and hexadecimal; characters, sound, images, compression

**Half Term 3**

* Python coding: Lists and mini project
* [Computer networks](https://www.bbc.co.uk/bitesize/guides/zvspfcw/revision/1), [connections and protocols](https://www.bbc.co.uk/bitesize/guides/zr3yb82/revision/1)

**Half term 4**

* Python coding: working with files; [SQL databases](https://classroom.thenational.academy/units/databases-and-sql-73d9)
* [Network security](https://www.bbc.co.uk/bitesize/guides/zj89dxs/revision/1)
* [Systems software](https://www.bbc.co.uk/bitesize/guides/zmqw7p3/revision/1)

**Half Term 5**

* Python coding: Dealing with files
* [Ethical, legal, cultural and environmental impact of digital technology](https://www.bbc.co.uk/bitesize/guides/zhx26yc/revision/1)

**Half Term 6**

* Python coding: Secret messages project
* Revision and mock exam preparation

Year 11:

**Half Term 1**

* [Computational thinking](https://www.bbc.co.uk/bitesize/guides/z4rbcj6/revision/1)
* [Searching and sorting algorithms](https://www.bbc.co.uk/bitesize/guides/zjdkw6f/revision/1)
* Using [flowcharts and pseudocode](https://www.bbc.co.uk/bitesize/guides/z6m7xfr/revision/1)

**Half Term 2**

* [Programming fundamentals](https://www.bbc.co.uk/bitesize/guides/znh6pbk/revision/1)
* [Sequence](https://www.bbc.co.uk/bitesize/guides/znh6pbk/revision/2) and [selection](https://www.bbc.co.uk/bitesize/guides/znh6pbk/revision/3)
* [Iteration](https://www.bbc.co.uk/bitesize/guides/znh6pbk/revision/5)
* [Arrays](https://www.bbc.co.uk/bitesize/guides/zb3yb82/revision/5)

**Half Term 3**

* Procedures and functions
* [Logic](https://www.bbc.co.uk/bitesize/guides/zjw8jty/revision/1) and [languages](https://www.bbc.co.uk/bitesize/guides/z6x26yc/revision/1)

**Half term 4**

* [Records and files](https://classroom.thenational.academy/units/programming-6-dictionaries-and-datafiles-b91f)

**Half Term 5**

* Revision and exam preparation

**Half Term 6**

* External examination period