

## Year 7 - Computing

### Areas of Learning

- **Algorithms:** Consider and apply aspects of computational thinking: problem decomposition, logical reasoning, pattern recognition, abstraction, pattern generalisation and algorithm design.
- **Programming & Development:** Use at least one graphical / visual programming language, and potentially one textual based programming language, to solve a variety of computational problems, using a variety of programming constructs, logic and Boolean operators.
- **Data & Data Representation:** Understand how computers store and manipulate data of different forms.
- **Hardware & Processing:** Understand the main functional units of a computer system, how data is generated, stored, processed by a computer, and how its output can be used.
- **Communication & Networks:** Appreciate the structure and operation of data communication networks, and how the internet works.
- **Information Technology:** Use standard tools to create and repurpose digital content. Use available tools to search for content and to judge content carefully in terms of its reliability.

### Approaches to learning

- Use online services that guide learners step-by-step, with interactive feedback, that allow them to learn a programming language.
- To undertake individual and team-based creative projects:
  - To build skills with common software packages that allow pupils to create and repurpose digital content.
  - To work on problem solving activities that promote the use and development of computational thinking and program development skills, within the context of cross-curricular topics and possibly related to real-world applications.
  - To use robotics or other electronic equipment to demonstrate the use of computer technology in measurement and control applications.
  - To simulate digital logic circuits in order to appreciate how circuitry in a computer operates.
  - To collection information and analyse it.
  - To research and investigate how people and organisations protect themselves, their identity and data online.
- Build simulations to demonstrate their understanding of a particular topic, such as the operation of a digital adder circuit.
- Document project work using Microsoft Word, process data using Microsoft Excel, and create presentations using Microsoft Powerpoint.
- To create poster displays of their work.
- Exercises and projects aimed at involving pupils in computational thinking.

## **Examples of learning**

- Pupils work through tasks that develop their skills with Microsoft Word, Excel and Powerpoint.
- Pupils create basic algorithms using flowcharting software, Raptor. Become familiar with its programming environment and work on more complex tasks that use more complicated logic / control structures.
- Pupils move on to develop programs using the Scratch environment, or similar.
- Pupils develop computer programs that instruct the movement of a robot around a course defined by a black line (line-following robot).
- Pupils research historical developments in computer technology to build a timeline regarding key technological developments. Group work leading to a poster display contributed to by the whole class.

## **References**

- P. Kemp, (2014), "Computing in the national curriculum: A guide for secondary teachers", ISBN: 978-1-78339-376-3.
- M. Dorling, et.al, (2014), "Computing progression pathways", Computing at School, download accessed on 07-07-2015 at [www.computingatschool.org.uk](http://www.computingatschool.org.uk)
- UK Department of Education, (2013), National Curriculum Documents available at: <https://www.gov.uk/government/publications/national-curriculum-in-england-computing-programmes-of-study/national-curriculum-in-england-computing-programmes-of-study#key-stage-2>, web-document published on 11 September 2013.