## COMPUTING - BIG PICTURE CURRICULUM PLANNING - KEY STAGE 3

### The big aims of KS3

- To enable the scientific and practical study of computation: what can be computed, how to compute it, and how programs can be written to solve problems
- To develop student understanding about how computers and telecommunications equipment work, including storage, retrieval, transmission and manipulation of different forms of data
- To develop in students the ability to be efficient and critical users of technology
- To develop understanding about the risks associated with the use of technology and how to keep themselves/others safe
- To develop and extend students’ subject/technical vocabulary
- To develop students’ technical understanding and competence so that they are able to create a range of digital artefacts which are fit for audience and purpose. Additionally, to select, use, manipulate and evaluate software and systems.
- To [inspire more girls](#) to study computing at KS4 and consider this as a career route

### Characteristics of a compelling learning experience

- Promotes learner curiosity, enjoyment, creativity, autonomy and resilience
- Provides engagement with ‘real-world’ audiences and problems
- Harnesses technology to develop collaborative work
- Engages learners with relevant, high-quality competitions, trips and speakers
- Fosters growth-mindset in learners
- Provides opps. to develop successful habits for learning, ie. ‘My FPHS’

### Key concepts

- **AB** - abstraction - the process of removing unnecessary detail from a problem in order to solve it
- **DE** - decomposition - the breaking down of problems into smaller parts
- **AL** - algorithmic thinking - the logical sequencing of steps to solve a problem
- **EV** - evaluation - judging outcomes to determine whether a solution/digital artefact is efficient, fit for purpose and audience
- **GE** - generalisation - recognising similarities and patterns in solutions in order to solve similar problems
# Key knowledge

- Computational thinking methods: Decomposition (breaking down), Abstraction (removing unnecessary detail), Pattern Recognition (spotting/using similarities) and Algorithmic Thinking (making steps & rules and representing these)
- Key constructs of programming; sequence, selection, iteration
- Programming language syntax (graphical - eg. Scratch/blockly and text based eg. Python)
- Boolean logic (AND OR and NOT) and how to present this
- Data types, data representation and Binary/Hex
- Features and purpose of a variety of software; graphics, modelling, online communications, programming, video/sound editing
- Digital literacy, computer legislation and ethics
- Collaborative working methods (ie. Google docs/drive/sharing)
- Features and purpose of different types of computer hardware and software
- Risks of technology (dangerous content/contact/conduct) for self, peers, others in digital spaces/communities, how to manage risk and report concerns,
- Features and benefits of a positive online profile/digital footprint
- The systems life-cycle ‘way of working’ (eg. analysis/design/implementation/testing/evaluation)

# Key skills

- Debugging
- Programming
- Applying boolean logic
- Applying computational thinking methods to design algorithms and solve problems
- Analysing and predicting
- Information handling: finding, creating, judging, manipulating data and information
- Applying the systems life cycle to any given computing/IT problem
- Technical proficiency