A picture containing logo

Description automatically generated

**01 April 2023**

Hope High School Computing CURRICULUM POLICY

|  |
| --- |
|  |

Computing Curriculum Policy

**Hope High School**

Carfield

Skelmersdale

Lancashire

WN8 9DP

Tel: 01695 721066

Curriculum Purpose.“Learning for Life”

Our curriculum should allow all learners to have parity of opportunity, be life ready, harness their potential, promote creativity, have rich experiences, and broaden their life choices. ​

At Hope High School the Curriculum is ambitious and tailored to meet the needs of ALL pupils. Pupils study a broad and balanced range of subjects up to Entry Level 1,2 and 3. At Key Stage 3 pupils have the opportunity to follow a knowledge rich curriculum in a wide range of subjects.

## Purpose

Our policy is intended to:

* Introduce the aims and objectives of the Computing Department.
* Outline the key components within Computing.
* Outline the knowledge skills and understanding for Key Stage 3.
* Explain the effective Teaching and Learning strategies utilised in Computing.
* Provide the formative and summative assessment strategies used within Computing.

## **Aims:**

Through Computing we want the pupils at Hope High School to be:

* able to understand and apply the fundamental principles and concepts of Computing, including abstraction, logic, algorithms and data representation.
* able to analyse problems in computational terms and have repeated practical experience of writing computer programs in order to solve such problems.
* able to evaluate and apply information technology, including new or unfamiliar technologies, analytically to solve problems.
* responsible, competent, confident and creative users of information and communication technology.

## Objectives:

To help meet the Aims of the Computing Curriculum the department will:

* Support pupils in the acquisition of knowledge, through the use of key concepts, terms, and vocabulary, providing opportunities to build a shared and consistent understanding. Glossaries, concept maps, and displays, along with regular recall and revision, can support this approach.
* Use supportive frameworks when planning lessons, such as PRIMM (Predict, Run, Investigate, Modify, Make and Use-Modify-Create. These frameworks are based on research and ensure that differentiation can be built in at various stages of the lessons.
* Bring abstract concepts to life with real-world, contextual examples and a focus on interdependencies with other curriculum subjects. This can be achieved through the use of unplugged activities, proposing analogies, storytelling around concepts, and finding examples of the concepts in pupils’ lives.
* Teach new concepts by first unpacking complex terms and ideas, exploring these ideas in unplugged and familiar contexts, then repacking this new understanding into the original concept. This approach, called ‘semantic waves’, can help pupils develop a secure understanding of complex concepts.
* Encourage collaboration, specifically using pair programming (ncce.io/qr03) and peer instruction, and also structured group tasks. Working together stimulates classroom dialogue, articulation of concepts, and development of shared understanding.
* When teaching programming, focus first on code ‘reading’ activities, before code writing. With both block-based and text-based programming, encourage pupils to review and interpret blocks of code. Research has shown that being able to read, trace, and explain code augments pupils’ ability to write code.
* When teaching programming, focus first on code ‘reading’ activities, before code writing. With both block-based and text-based programming, encourage pupils to review and interpret blocks of code. Research has shown that being able to read, trace, and explain code augments pupils’ ability to write code.
* Model processes or practices — everything from debugging code to binary number conversions — using techniques such as worked examples (ncce.io/qr02) and live coding. Modelling is particularly beneficial to novices, providing scaffolding that can be gradually taken away.
* Use physical computing and making activities that offer tactile and sensory experiences to enhance learning. Combining electronics and programming with arts and crafts (especially through exploratory projects) provides pupils with a creative, engaging context to explore and apply computing concepts.
* Use formative questioning to uncover misconceptions and adapt teaching to address them as they occur. Awareness of common misconceptions alongside discussion, concept mapping, peer instruction, or simple quizzes can help identify areas of confusion.
* Provide activities with different levels of direction, scaffolding, and support that promote active learning, ranging from highly structured to more exploratory tasks. Adapting your instruction to suit different objectives will help keep all pupils engaged and encourage greater independence.
* Use a variety of activities to consolidate knowledge and understanding of the function and structure of programs), including debugging, tracing, and Parson’s Problems. Regular comprehension activities will help secure understanding and build connections with new knowledge.

# Subject Content

## Key Stage 3

A high-quality computing education equips pupils to use computational thinking and creativity to understand and change the world. Computing has deep links with mathematics, science, and design and technology, and provides insights into both natural and artificial systems. The core of computing is Computing, in which pupils are taught the principles of information and computation, how digital systems work, and how to put this knowledge to use through programming. Building on this knowledge and understanding, pupils are equipped to use information technology to create programs, systems and a range of content. Computing also ensures that pupils become digitally literate – able to use, and express themselves and develop their ideas through, information and communication technology – at a level suitable for the future workplace and as active participants in a digital world. During Computing, pupils should be taught to:

* design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems.
* understand several key algorithms that reflect computational thinking [for example, ones for sorting and searching]; use logical reasoning to compare the utility of alternative algorithms for the same problem.
* use two or more programming languages, at least one of which is textual, to solve a variety of computational problems; make appropriate use of data structures [for example, lists, tables or arrays]; design and develop modular programs that use procedures or functions.
* understand simple Boolean logic [for example, AND, OR and NOT] and some of its uses in circuits and programming; understand how numbers can be represented in binary, and be able to carry out simple operations on binary numbers [for example, binary addition, and conversion between binary and decimal].
* understand the hardware and software components that make up computer systems, and how they communicate with one another and with other systems.
* understand how instructions are stored and executed within a computer system; understand how data of various types (including text, sounds and pictures) can be represented and manipulated digitally, in the form of binary digits.
* undertake creative projects that involve selecting, using, and combining multiple applications, preferably across a range of devices, to achieve challenging goals, including collecting and analysing data and meeting the needs of known users.
* create, re-use, revise and re-purpose digital artefacts for a given audience, with attention to trustworthiness, design and usability.
* understand a range of ways to use technology safely, respectfully, responsibly and securely, including protecting their online identity and privacy; recognise inappropriate content, contact and conduct and know how to report concerns.

# Teaching and Learning

Good quality teaching and learning is at the heart of improved life chances for the pupils at Hope High School. Our pupils benefit from consistency in their lives. Hope High has adopted the teaching and learning model of Rosenshine’s 10 principles of Instruction. These principles will be evident in classrooms daily.

In Computing the principles are:

* Daily review. Lessons will begin with a review of learning from previous lessons. This is to support our pupils cognitive load. This could be a review of new tools used or a re-cap on a new process.
* Present new material using small steps. Teacher demonstration is an integral part of the delivery of the subject. New concepts and skills will be introduced in small steps using live or video demonstration.
* Ask questions. A range of interactive questioning techniques will be used to check pupil understanding. Direct questioning, use of interactive whiteboards and NearPod are just a few of the techniques used to check pupil understanding.
* Provide models. Example projects/visual aids will be provided to support pupil understanding.
* Guide Student practice. Teachers, TA’s and the school technician will be used to help and support pupils in lessons with an emphasis on encouraging pupil independence.
* Check for student understanding. Formative assessment techniques such as direct questioning of individual pupils and observation of skills and techniques will be used to check knowledge and understanding.
* Obtain a high success rate. Pupils will be encouraged to practise techniques and skills until mastery is achieved in that area.
* Provide scaffolds for difficult tasks. In Computing the support of the teacher, TA or Technician is one of the main scaffolds that can support our students in difficult tasks. An individual demonstration, support with a practical process or a verbal instruction can help pupils move forward with their learning in Computing.
* Independent practice. Opportunities will be provided to pupils for them to apply their knowledge, skills and understanding to different computing scenarios. Independence will be supported and encouraged in all our pupils in a safe and calm environment.
* Weekly and monthly review. The use of OneNote is used as evidence to keep and monitor pupils work to help demonstrate progress in Computing. Pupils will revisit skills they used previously to help consolidate and support learning.

# Assessment

At Hope High we firmly believe that assessment should be used as a tool to help move pupil learning forward. The day to day, lesson by lesson, formative assessment that takes place with pupils will help them make progress. This will inform teachers of areas of mastery and support them in the development of scaffolding materials to help pupils who need that extra support to achieve mastery.

This formative assessment coupled with end of topic rubric or summative assessments will provide the teacher with a holistic overview of a pupil progress and inform the grade for termly reports that are sent to parents/carers.

In Computing pupils are assessed on the knowledge and understanding they have of the ten strands of Computing.

The ten strands of Computing are:

* Algorithms
* Computer Networks
* Computer Systems
* Creating Media
* Data and information
* Design and development
* Effective use of tools
* Impact of technology
* Programming
* Safety and security

Assessments are made clear to learners to develop their independence and evaluation of their work. This will inform future target setting. Self-evaluation is key in getting learners to understand what they can do well and what they need to improve on further.

# Monitoring

## The Head of School and leadership team will:

Monitor the subject through the Hope High self-evaluation schedule and monitoring schedule which are reviewed annually.

## Departmental leader will:

* Monitor learners work and quality of teaching and learning.
* Review Curriculum Maps and Schemes of Work based on suitability of use.
* Analyse pupil performance data.
* Attend link meetings.

## Links to other policies:

• Teaching and Learning

• Behaviour for Learning

• Monitoring

• Assessment for Learning

• Health and Safety

• Marking and Feedback

|  |  |
| --- | --- |
| Date Approved: |  |
| Review date: |  |
| Signed subject Lead: |  |
| Signed Headteacher: |  |

A picture containing text, clipart

Description automatically generated