The Good Shepherd Catholic Primary School

Following Jesus,
The Good Shepherd,
in all we say and do

Our Computing Curriculum

OUR LADY IMMACULATE
CATHOLIC ACADEMIES TRUST

Computing: Intent

At The Good Shepherd our computing curriculum ensures a high-quality computing education in order to prepare children for their future by giving them the opportunities to gain knowledge and develop skills that will equip them for an ever-changing digital world. Appropriate knowledge and understanding of Information technology are of increasing importance for children's future both in a future career and at home especially in the current climate!

We focus on a progression of skills in digital literacy, computer science and information technology to ensure that children become competent in using and understanding technology. We also recognise the critical importance of children understanding how to keep themselves and others safe online.

We want children to become autonomous, independent users of computing technologies, gaining confidence and enjoyment from their activities. We want the use of technology to support learning across the entire curriculum and to ensure that our curriculum is accessible to every child. Not only do we want them to be digitally literate and competent end-users of technology but through our computer science lessons we want them to develop creativity, resilience and problem-solving and critical thinking skills. We want our pupils to have a breadth of experience to develop their understanding of themselves as individuals within their community but also as members of a wider global community and as responsible digital citizens.

The national curriculum for Computing aims to ensure that all pupils:

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

At The Good Shepherd, we use the National Centre for Computing Education (NCCE) curriculum and resources as a basis for providing a clear and comprehensive scheme of work which meets the requirements of the National Curriculum.

Children in all year groups are exposed to a range of topics which encourage progression across the key strands of computer science, digital literacy and information technology. All children have access to the hardware and software needed to develop knowledge and skills of digital systems and their applications.

Our new IT Suite is equipped with new Chromebooks. In addition to this, children have access to further technologies such as programmable toys (BeeBots) all other up to date equipment which we loan from Northants computing hub, who we have a good working relationship with.. This equipment is used to ensure children have ample opportunities to apply their learning across the curriculum.

Online Safety is key in all aspects of computing and through these lessons, the children have the opportunity to explore and respond to key issues such as digital communication, cyberbullying, online safety, security, plagiarism and social media.

At The Good Shepherd Catholic Primary School, we strive to nurture and develop every child's interest and enjoyment of Computing. The teaching of Computing is structured and carefully planned at every stage.

We have worked hard to perfect our curriculum to meet the needs of our children. As children grow in basic skills with technology, we wanted to make sure that they were given a challenge. The NCCE framework is perfect for us as it develops the basic skills of the children and it helps us to ensure that all of the children leave with the skills in which they need for the future. We have used this to create long term plans for each year group so that teachers can see the progression from previous learning and what will be expected in the future.

Each unit builds on the skills which have been previously taught and enables the children to build their knowledge in computing as they move up the school. This enables children to become advanced in the skills in computing across the three aspects of computing and it also enables teachers to build on prior knowledge. Children must master all of those skills in their year group in order to progress to the next stage - this enables us to keep high standards for computing.

- Assessment in Computing is ongoing and evidence in a class floor book (see making policy for more information) the children's progress will be updated by the class teacher each term.
- Progress is monitored by the Computing curriculum lead throughout the year.
- All lessons PowerPoints and files required for learning activities and assessment can be found on the Teach Computing, which is found at https://teachcomputing.org/curriculum

Lead with concepts

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.

Unplug, unpack, repack

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 (semantic waves) can help pupils develop a secure understanding of complex concepts.

Create projects

Use project-based learning activities to provide pupils with the opportunity to apply and consolidate their knowledge and understanding. Design is an important, often overlooked aspect of computing. Pupils can consider how to develop an artefact for a particular user or function, and evaluate it against a set of criteria.

Challenge misconceptions

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.

Structure lessons

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 lesson.

Work together

Encourage collaboration, specifically using pair programming and peer instruction, and also structured group tasks. Working together stimulates classroom dialogue, articulation of concepts, and development of shared understanding.

Model everything

Model processes or practices — everything from debugging code to binary number conversions — using techniques such as worked examples and live coding. Modelling is particularly beneficial to novices, providing scaffolding that can be gradually taken away.

Add variety

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.

Make concrete

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.

Read and explore code first

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.

Get hands-on

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.

Foster program comprehension

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.

Find out more about our principles and add some or all to your personal pedagogy toolkit.

ncce.io/pedagogy