

**Our children are receptive, inquisitive learners who, through our Gospel values, have a unique sense of the world**

**The Computing Curriculum K&S at St Teresa’s Catholic Academy – Lower Key Stage 2**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NC Objective    Pupils should be taught to: | Year 3 | | Year 4 | |
| Skills | Knowledge | Skills | Knowledge |
| **Computer science**   Design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts. | Turn a simple real life situation into an algorithm. Design an  algorithm and turn it into code.  Identify errors in programs they have  created and fix it. | Know how to deconstruct a program into  manageable parts.  Know how to correct an error in an algorithm. | Turn a real life situation into an  algorithm using a  design that shows how it can be  accomplished in code.  Identify errors in code using different methods. | Know how to turn real-life situations  into an algorithm using coding  structures for selection and  repetition.  Know how to debug their own programs. |
| **Computer science**   Use sequence, selection, and repetition in programs; work with variables and various forms of input and output. | Design and code a program that  follows a simple  sequence. Experiment with timers and repeat commands. Use  variables to store information. | Know how to use timers in programs  and understand the difference in the effect of using a timer command  rather than a repeat command when  creating repetition effects. Know how  variables can be used to store  information while a program is executing. | Use repetition in code, for example using a loop that continues until a condition is met such as the correct  answer being  answered. Use timers accurately to create  repetition effects.  Use selection in programming, for  example using an ‘if statement’ for a  question and the  program takes one of two paths. Use  variables and change the values of the  variables. Use input  and output features such as ‘print to screen’. | Know how to use timers to achieve repetition.  Understand ‘if  statements’ and how they can be used to  achieve the effects  in program design.  Understand how variables can be used to store information while a program is  executing. Know how to make use of user inputs and outputs. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Computer science**   Use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs | Design programs using logical  achievable steps. Identify ‘if’ statements,  repetition and  variables. Read programs with  several steps and make predictions. | Know how to structure programs logically. Know how  to make attempts to  ‘step through’ more complex code in  order to identify  errors in algorithms and know how to  correct this. Know how to read  programs with several steps and predict the outcome accurately. | Identify errors in code using different  methods, such as stepping through lines of code and fixing them.  Read programs containing several  steps and predict the outcomes with increasing accuracy. | Understanding how to show they are thinking of the structure of a  program in logical  achievable steps and developing an  awareness of ‘if statements’,  repetition and variables.  Know how to trace code using step- through methods to identify errors in codes and how to  correct this logically.  Know how to read programs with  several steps and  how to predict the outcome accurately |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Computer science**   Understand computer networks, including the internet; how they can provide multiple services, such as the World Wide Web, and the opportunities they offer for communication and collaboration. | Identify different ways in which the  internet can be used  for communication.  Open an email, respond  appropriately and attach files. | Know the ways that the internet can be used to provide  different methods of communication.  Know how to open, respond and attach files to emails. | Recognise the main component parts of  hardware that allows  computers to join and form a network.  Identify that network and  communication  components can be found in many  different devices  which allows them to join the internet. | Know the main component parts of hardware which  allows computers to join and form a  network. Understand the online safety implications  associated with the  ways the internet can be used to provide  different methods of communication. |
| **Information technology**   Use search technologies effectively, appreciate how results are selected and ranked, and be discerning in evaluating digital content. | Carry out searches to find digital  content on a range of online systems,  such as within Purple  Mash or an internet search engine. | Understand how to carry out searches to retrieve digital content using the internet. | Explain the purpose of a search engine and the main  features within it.  Look at information on a webpage and make predictions  about the accuracy of the information contained within it. | Understand the function, features and layout of a  search engine. Know how to appraise  selected webpages for credibility and information at a basic level. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Information technology**   Select, use and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems and content that accomplish given goals, including collecting, analysing, evaluating and presenting data and information | Collect data and put it into software.  Analyse data using features within  software, such as formula in  spreadsheets (2Calculate). Present data and information using different  software. Consider the most  appropriate  software to use to complete a task.  Create content and attach this to emails. | Know how to collect, analyse, evaluate  and present data and information  using a selection of software, e.g. using a branching database  (2Question). | Create and improve solutions to a  problem based on feedback. Review  solutions that others have created using a  checklist or criteria.  Work collaboratively to create content  and solutions. Share  digital content using a variety of applications. | Know how to make improvements to digital solutions  based on feedback.  Know how to make informed software choices when presenting  information and data.  Understand how to create linked content using a range of software.  Know how to share digital content with others. |
| **Digital Literacy**   Use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact. | Create a secure and private password.  Explain the consequences of not  keeping passwords secure. Behave  respectfully and safely online, reporting  unacceptable content. | Know the importance of having a secure  passwords and why this should not be  shared with anyone else. Understand  the importance of  staying safe and the importance of their own conduct. Know more than one way to report  unacceptable  content and contact. | Demonstrate a good understanding of  online safety rules. Explain the right to privacy both on and offline. Recognise  that wellbeing can be affected by  technology use.  Understand how to report any concerns with content and  contact online and use immediate strategies to keep safe. | Explore key concepts relating to online  safety using concept mapping, such as 2Connect.  Know how to help others to understand the importance of  online safety. Know a range of ways of reporting  inappropriate content and contact. |