St George's School
Computer Science
KS5 Curriculum

PREREQUISITE KNOWLEDGE & SKILLS

The foundations needed to thrive in this subject.

Who should study this subject?
Pupils who have a keen interest in computing and are comfortable with the subject material covered during KS4.

Key Skills developed during KS4:

- ➢ understanding the fundamental principles and concepts of Computer Science.
- ➢ analysing problems by designing, writing and debugging programs.
- ➢ thinking creatively, innovatively, analytically, logically and critically.
- ➢ understanding the components that make up digital systems.
- ➢ understanding the impacts of digital technology to the individual and to wider society.

St George's course entry requirements:
Grade 5 in GCSE(9-1) Mathematics.

QUALIFICATION

Exam Board, aims and objectives.

A Level Computer Science, OCR H446

The aims of this qualification are to enable learners to develop:

- ➢ An understanding and ability to apply the fundamental principles and concepts of computer science. This includes abstraction, decomposition, logic, algorithms and data representation.
- ➢ The ability to analyse problems in computational terms through practical experience of solving such problems, including writing programs to do so.
- ➢ The capacity to think creatively, innovatively, analytically, logically and critically.
- ➢ The capacity to see relationships between different aspects of computer science.

ASSESSMENT

Internal monitoring and final assessment.

Internal Assessment:
Formal prep is assigned once a week and is monitored for completion and accuracy. Informal prep is for pupils to further their NEA Programming Project using the skills they acquire throughout the course. This is monitored for progress.
Check-point assessments take place after every topic.
Students complete mock exams in Year 12 and Year 13.

Final assessment:
1 written examination paper on Computer Systems (40%)
1 written examination paper on Algorithms and Programming (40%)
NEA Programming Project (20%)

Each written paper is 2 hours 30 minutes long and is non-calculator.
The NEA Programming Project is assessed by the teacher and moderated by the exam board.

ENRICHMENT

Trips & Visits, wider reading, etc.

Visits and Events:

Bebras Challenge https://www.bebras.uk/

National Cipher Challenge https://www.cipherchallenge.org/

Wider reading:

Isaac Computer Science https://isaaccomputerscience.org/

Harvard University https://pll.harvard.edu/subject/computer-science

Nottingham University https://www.youtube.com/user/Computerphile

NEXT STEPS *Where this subject can take you.*	**Related University Courses:** A wide variety of university degree courses are available in the fields of computer science, information systems, software engineering, artificial intelligence and health informatics. Computer science related fields are also offered in many joint degrees. **Career Paths:** Computing is a huge industry and offers many careers including Software Developer, Applications Programmer, Systems Programmer, Multimedia Programmer, Systems Analyst, Computer Sales Support, Database Administrator, IT Technical Support Officer, Computer Security Consultant, Games Developer, IT Consultant, and Web Designer. There are also many careers where computer science knowledge can be very advantageous.

	Year 12
Autumn Term	**Topics:** Binary; Data Representation; Legislation; Ethical Issues; Environmental Issues; Cultural Issues; Boolean Algebra; Computational Thinking. **Skills:** Pupils start with binary as this is fundamental to all computing. This topic includes the following skills. ➢ Establishing how binary numbers can be made from digital bits. ➢ Doing calculations using binary shifting and binary arithmetic. ➢ Converting to and from denary and hexadecimal numbers. ➢ Discovering how binary numbers can be used to represent things in a computer. ➢ Using ASCII and Unicode character sets. Pupils learn about broader technological issues. These are discussed early in the course so pupils can then assimilate relevant knowledge from various sources for the duration of the course. These topics include the following skills. ➢ Evaluating which Acts of Parliament apply to a given scenario. ➢ Evaluating the ethical implications of new technology. ➢ Evaluating the impact of new technology on cultural groups. ➢ Evaluating the impact of new technology on the environment. Pupils then move on to boolean algebra, as this topic follows on from the work they have already done on binary. This topic includes the following skills. ➢ Using logic gates to make electronic circuits. ➢ Evaluating and simplifying boolean expressions. ➢ Using Karnaugh maps to simplify boolean expressions. ➢ Using adder circuits to do binary addition. ➢ Using flip flop circuits to store binary information. Pupils learn about computational thinking, which provides an overview of the techniques that can be used to solve problems using computers. This information should be beneficial to pupils as they begin their NEA programming project. This topic includes abstraction, decomposition, thinking ahead and problem solving. **Assessment:** Formal prep is assigned once a week and is monitored for completion and accuracy. Informal prep is for pupils to further their NEA Programming Project using the skills they acquire throughout the course. Check-point assessments take place after every topic.
Spring Term	**Topics:** Software Development; Data Types; Programming. **Skills:** Pupils learn about different software development methodologies and how algorithms can be constructed. Pupils will utilise this knowledge in their NEA Programming Project.

	Pupils learn how negative and decimal point numbers can be represented in a computer. This topic includes the following skills. ➢ Representing negative numbers in sign and magnitude format. ➢ Representing negative numbers in two's complement format. ➢ Representing decimal point numbers in fixed point format. ➢ Representing decimal point numbers in floating point format. ➢ Adding and subtracting floating point numbers. Programming is a major component of the course and is critical for the NEA Programming Project. Pupils learn the following fundamental skills. ➢ Utilising variables, constants and data types. ➢ Utilising input/output, string manipulation and typecasting. ➢ Utilising arithmetic operators, comparison operators and boolean operators. ➢ Utilising sequence, selection, and iteration to build programs. Pupils then proceed to learn more advanced programming techniques. ➢ Utilising subroutines, parameters and arguments. ➢ Utilising subroutines recursively. ➢ Utilising object oriented programming. ➢ Utilising bitwise manipulation for real-world scenarios. **Assessment:** Formal prep is assigned once a week and is monitored for completion and accuracy. Informal prep is for pupils to further their NEA Programming Project using the skills they acquire throughout the course. Check-point assessments take place after every topic.
Summer Term	**Topics:** Data Structures; **Skills:** Pupils learn about abstract data types and data structures. Pupils learn the following skills. ➢ Utilising arrays, tuples, records, and lists. ➢ Utilising linked lists, stacks, queues, graphs, trees, and hash tables. ➢ Traversing data structures. **Assessment:** Mock examination - 2 hours non-calculator. Formal prep is assigned once a week and is monitored for completion and accuracy. Informal prep is for pupils to further their NEA Programming Project using the skills they acquire throughout the course. Check-point assessments take place after every topic.

	Year 13
Autumn Term	**Topics:** Algorithms; Web Technology. **Skills:** Algorithms are a very important feature of computing that are used in secondary education, higher education and industry. Pupils learn the following skills. ➢ Analysing and designing algorithms. ➢ Evaluating the suitability of algorithms for a given task and data set. ➢ Measuring the efficiency of algorithms using Big-O notation. ➢ Comparing the complexity of algorithms. ➢ Utilising standard algorithms including linear search, binary search, bubble sort, insertion sort, merge sort, quicksort, Dijkstra's shortest path, and A* algorithm. Web technology is a popular topic given the success of the internet. Pupils learn the following skills. ➢ Utilising HTML, CSS and JavaScript to make web pages. ➢ Making calculations with the PageRank algorithm and learning about search engine indexing. ➢ Utilising client and server side processing.

	Assessment: Formal prep is assigned once a week and is monitored for completion and accuracy. Informal prep is for pupils to further their NEA Programming Project using the skills they acquire throughout the course. Check-point assessments take place after every topic.
Spring Term	**Topics:** Compression, Encryption and Hashing; Databases; Networks. **Skills:** Compression, encryption and hashing are solutions to critical problems that arise in computing. Pupils learn the following skills. ➢ Applying compression algorithms to given scenarios. ➢ Making communication secure using encryption algorithms. ➢ Utilising hashing techniques. Databases are a crucial part of most software. Pupils learn the following skills. ➢ Utilising flat files and relational databases to store data. ➢ Applying methods to capture, manage and exchange data. ➢ Writing Structured Query Language (SQL) to access and manipulate databases. ➢ Utilising transaction processing to maintain the referential integrity of a database. Networks are an essential element in today's interconnected world. Pupils learn the following skills. ➢ Utilising protocols and standards to implement communication between computers. ➢ Investigating the structure of the internet. ➢ Evaluating network security and threats. ➢ Evaluating client-server and peer to peer networks. **Assessment:** Mock examination - 2 hours non-calculator. Formal prep is assigned once a week and is monitored for completion and accuracy. Informal prep is for pupils to further their NEA Programming Project using the skills they acquire throughout the course. Check-point assessments take place after every topic.
Summer Term	**Topics:** All formal teaching of the specification is complete by the Spring term. This term is used for revision and completion of the NEA Programming Project. **Skills:** Students are encouraged to do the following. ➢ Develop independent revision skills. ➢ Secure subject knowledge using spaced repetition and active recall. ➢ Complete past papers and examine the mark schemes in detail. ➢ Review good exam technique. ➢ Test each other in small groups. **Assessment:** Final Assessment as detailed above.