

# Roundwood Park School



## KS5 Computer Science (OCR)

The aims and objectives of A-level Computer Science are to enable students to:

- Develop further an interest and understanding of computer science
- Develop a solid foundation of tier 3 language coding using OOP paradigms
- Emphasis on the mathematical skills used to express computational laws and processes
- Understand the fundamental algorithms that are involved in data processing and decision making
- Acquire a range of problem solving and analytical skills that are applicable to employment in a range of fields both technical and non-technical roles
- Develop all aspects of ASPIRE and help develop the Roundwood leaver

In A-Level computer science, the order the curriculum is taught in has been carefully considered to build on content and understanding developed at KS4, avoiding unnecessary repetition whilst also ensuring that students new to the subject are appropriately supported. The course consists of three main areas of study, two of which are assessed externally and the third, the programming project, is a piece of coursework which is external assessed. This gives the students a chance to showcase their skills and understanding in a practical way.

### Assessment objectives:

**AO1:** Demonstrate knowledge and understanding of the principles and concepts of computer science, including abstraction, logic, algorithms and data representation

**AO2:** Apply knowledge and understanding of the principles and concepts of computer science including to analyse problems in computational terms

**AO3:** Design, program and evaluate computer systems that solve problems, making reasoned judgements about these and presenting conclusions

Component	% of A Level Computer Science (H446)			
	AO1	AO2	AO3	Total
Computer systems (H446/01)	21*	9*	10*	40
Algorithms and programming (H446/02)	15*	18*	7*	40
Programming project (H446/03 or H446/04)	0*	3*	17*	20
Total	35*	30*	35*	100

## Rationale

The plan is based on:

- Requirements of the Paper 1 and Paper 2 syllabus
- Allowing students to work independently to complete their NEA Systems Analysis and Programming Assignment
- Feed in from prior knowledge developed from GCSE Computer Science
- Ties in with teacher's previous experience from working in industry

Year / term	Unit of work: Teacher A	Unit of work: Teacher B	Assessment
Year 12 Autumn Term	<b>Focus: COMP 1</b> <b>Section 6: Data Types</b> <ul style="list-style-type: none"><li>• Primitive data types, binary and hexadecimal</li><li>• ASCII and Unicode</li><li>• Binary Arithmetic</li><li>• Floating point Arithmetic</li><li>• Bitwise manipulation and masks</li></ul>	<b>Focus: COMP 2</b> <b>Section 11: Programming Techniques</b> <ul style="list-style-type: none"><li>• Programming Basics</li><li>• Selection</li><li>• Iteration</li><li>• Subroutines and Recursion</li><li>• Use of an IDE</li><li>• Use of Object Orientated Programming</li></ul>	Weekly interleaved learning.  End of unit assessments
	<b>Focus: COMP 1</b> <b>Section 1: Components of a computer</b> <ul style="list-style-type: none"><li>• Processor components</li><li>• Processor performance</li><li>• Types of processor</li><li>• Input devices</li><li>• Output devices</li><li>• Storage devices</li></ul>	<b>Focus: COMP 2</b> <b>Section 7: Data Structures</b> <ul style="list-style-type: none"><li>• Arrays, Tuples and Records</li><li>• Queues</li><li>• Lists and Linked Lists</li><li>• Stacks</li><li>• Hash Tables</li><li>• Graphs</li><li>• Trees</li></ul>	Weekly interleaved learning.  End of unit assessments

<b>Super Curricular</b>	<b><u>Read:</u></b> Daniel Hillis, W (1999) The Pattern On The Stone: The Simple Ideas That Make Computers Work; Kubica, J (2012) Computational Fairy Tales <b><u>Watch:</u></b> <b><u>Research:</u></b>	<b><u>Read</u></b> Kubica, J (2012) Computational Fairy Tales <b><u>Watch:</u></b> <b><u>Research:</u></b>	

Year / term	Unit of work: Teacher A	Unit of work: Teacher B	Assessment
Year 12 Spring Term	<b>Focus: COMP 1</b> <b>Section 2: System Software</b> <ul style="list-style-type: none"> <li>• Functions of an Operating System</li> <li>• Types of Operating System</li> <li>• The nature of applications</li> <li>• Programming language translators</li> </ul>	<b>Focus: COMP 2</b> <b>Section 12: Algorithms</b> <ul style="list-style-type: none"> <li>• Analysis and Design of Algorithms</li> <li>• Searching Algorithms</li> <li>• Bubble sort and insertion sort</li> <li>• Merge sort and quick sort</li> <li>• Graph Traversals</li> <li>• Optimisation Algorithms</li> </ul>	Weekly interleaved learning.  End of unit assessments
	<b>Focus: COMP 1</b> <b>Section 3: Software Development</b> <ul style="list-style-type: none"> <li>• Systems analysis methods</li> <li>• Writing and following algorithms</li> <li>• Programming paradigms</li> <li>• Assembly Language</li> </ul>		Weekly interleaved learning.  End of unit assessments
Super Curricular	<u>Read:</u> Kubica, J (2012) Computational Fairy Tales <u>Watch:</u> Source Code (2011); <u>Research:</u>	<u>Read</u> Kubica, J (2012) Computational Fairy Tales; Christian, B (2016) Algorithms to live by; <u>Watch:</u> The Secret Rules Of Modern Algorithms (2015) <u>Research:</u>	

Year / term	Unit of work: Teacher A	Unit of work: Teacher B	Assessment
Year 12 Summer Term	<b>Focus: COMP 2</b> <b>Section 4: Exchanging Data</b> <ul style="list-style-type: none"> <li>• Compression, encryption and hashing</li> <li>• Database concepts</li> <li>• Relational Databases and normalisation</li> <li>• Introduction to SQL</li> <li>• Defining and updating tables using SQL</li> <li>• Transaction processing</li> </ul>	<b>Focus: COMP 3</b> <b>Programming Project</b> <ul style="list-style-type: none"> <li>• Analysis</li> <li>• Design</li> </ul>	Weekly interleaved learning.  End of unit assessments
			Weekly interleaved learning.  End of unit assessments
Super Curricular	<u>Read:</u> Kubica, J (2012) Computational Fairy Tales <u>Watch:</u> <u>Research:</u>	<u>Read</u> Kubica, J (2012) Computational Fairy Tales <u>Watch:</u> <u>Research:</u>	

Year / term	Unit of work: Teacher A	Unit of work: Teacher B	Assessment
Year 13 Autumn Term	<b>Focus: COMP 1</b> <b>Section 5: Networks and web technologies</b> <ul style="list-style-type: none"> <li>• Structure of the internet</li> <li>• Internet communication</li> <li>• HTML and CSS</li> <li>• Web forms and Javascript</li> <li>• Search Engine Indexing</li> <li>• Client server and peer to peer</li> </ul>	<b>Focus: COMP 1</b> <b>Section 8: Boolean Algebra</b> <ul style="list-style-type: none"> <li>• Logic Gates and truth tables</li> <li>• Simplifying Boolean Expressions</li> <li>• Karnaugh Maps</li> <li>• Adders and D-type flip-flops</li> </ul>	Weekly interleaved learning.  End of unit assessments
	<b>Focus: COMP 1</b> <b>Section 9: Legal, Moral, Ethical and Cultural Issues</b> <ul style="list-style-type: none"> <li>• Computing related legislation</li> <li>• Ethical, Moral and Cultural issues</li> <li>• Privacy and censorship</li> </ul>	<b>Focus: COMP 3</b> <b>Programming Project</b> <ul style="list-style-type: none"> <li>• Development</li> <li>• Testing</li> <li>• Evaluation</li> </ul>	Weekly interleaved learning.  End of unit assessments
Super Curricular	<u>Read:</u> Daniel Hillis, W (1999) The Pattern On The Stone: The Simple Ideas That Make Computers Work Kubica, J (2012) Computational Fairy Tales <u>Watch:</u> Google and the world brain (2013); ex_machina (2014); Snowden (2016), Terms and Conditions may apply (2013), Do you trust this computer? (2018) <u>Research:</u>	<u>Read</u> Kubica, J (2012) Computational Fairy Tales <u>Watch:</u> <u>Research:</u>	

Year / term	Unit of work: Teacher A	Unit of work: Teacher B	Assessment
Year 13 Spring Term	<b>Focus: COMP 1</b> <b>Exam Practice and Interleaved learning</b> <ul style="list-style-type: none"> <li>Section 1: Components of a Computer</li> <li>Section 2: Systems Software</li> <li>Section 3: Software Development</li> <li>Section 5: Networks and web technologies</li> </ul>	<b>Focus: COMP 2</b> <b>Exam Practice and Interleaved learning</b> <ul style="list-style-type: none"> <li>Section B of Comp 2 Exam on OOP questions</li> <li>Section 7: Data Structures</li> <li>Section 12: Algorithms</li> </ul>	Weekly interleaved learning.  Timed Exam Questions
		<b>Focus: COMP 3</b> <b>Programming Project</b> <ul style="list-style-type: none"> <li>Marking and moderation</li> </ul>	Weekly interleaved learning.  Timed Exam Questions
Super Curricular	<u>Read:</u> Daniel Hillis, W (1999) The Pattern On The Stone: The Simple Ideas That Make Computers Work Kubica, J (2012) Computational Fairy Tales <u>Watch:</u> <u>Research:</u>	<u>Read</u> Kubica, J (2012) Computational Fairy Tales <u>Watch:</u> <u>Research:</u>	

Year / term	Unit of work: Teacher A	Unit of work: Teacher B	Assessment
Year 13 Summer Term	<b>Focus: COMP 1</b> <b>Exam Practice and Interleaved learning</b> <ul style="list-style-type: none"> <li>Exam language</li> </ul>	<b>Focus: COMP 2</b> <b>Exam Practice and Interleaved learning</b> <ul style="list-style-type: none"> <li>Exam language</li> </ul>	Weekly interleaved learning.
			Timed Exam Questions
Super Curricular	<u><b>Read:</b></u> Daniel Hillis, W (1999) The Pattern On The Stone: The Simple Ideas That Make Computers Work Kubica, J (2012) Computational Fairy Tales <u><b>Watch:</b></u> <u><b>Research:</b></u>	<u><b>Read</b></u> Kubica, J (2012) Computational Fairy Tales <u><b>Watch:</b></u> <u><b>Research:</b></u>	Weekly interleaved learning.
			Timed Exam Questions