

## So, you've chosen to study GCE Advanced Level Computer Science?

A very wise choice! Computer Science is an ever-changing subject that will prepare for almost any field of work. The A-level will have a focus on the internal components of a computer system, starting with the very foundations of how the binary number system is used as instructions to a processor. You will also consider external factors to a computer system, such as the different types of networks used to connect computers, as well as a study on the Internet and how web-design has evolved over the years. Most importantly, you will learn how to develop systems yourself through learning a new programming language (VB.NET) and consider the similarities / differences between modern programming languages. You will then build on these programming skills in Year 13 by developing a program of your choice for coursework, along with test criteria to prove if it works.

Firstly, it is not vital if you haven't studied GCSE Computer Science before, what is important is you have good experience in problem-solving and logical puzzles. Experience in Maths is also beneficial as many topics make use of mathematical concepts such as Floating Point Notation and Boolean Algebra.

<p><b>Preparation</b></p>	<ul style="list-style-type: none"> <li>✓ <b>Keep interested in your studies.</b> The more you read into topics about technology, the more questions you are likely to have. <b>This is learning</b>, as you are developing a curiosity for how modern computing issues are solved and will investigate how it works. New technologies are being developed at a rapid pace, so feel free to read beyond the subject into topics such as Quantum Computing, Non-Standard Computation, Processors or Virtual Reality.</li> <li>✓ <b>Contemporary issues</b> – or ‘What’s in the news?’ Keeping up with current affairs is a great way to gain depth and breadth of knowledge relevant to your studies. Clearly at the moment, the impact of the coronavirus is dominating news stories – look into this from a Computer Science viewpoint:             <ul style="list-style-type: none"> <li>○ <b>EpiRisk</b> – A program which is trying to predict the spread of coronavirus by mining large amounts of data from online sources (websites, social media etc.). It also simulates how preventions can help reduce the spread – <a href="https://spectrum.ieee.org/the-human-os/biomedical/devices/predicting-the-coronavirus-next-moves">https://spectrum.ieee.org/the-human-os/biomedical/devices/predicting-the-coronavirus-next-moves</a></li> <li>○ <b>Machine Learning</b> – look into the concept of “machine learning” and how computers could theoretically learn new skills – <a href="https://www.simplilearn.com/tutorials/machine-learning-tutorial/what-is-machine-learning">https://www.simplilearn.com/tutorials/machine-learning-tutorial/what-is-machine-learning</a></li> </ul> </li> <li>✓ <b>See the short term focus below.</b> Also, keep on top of subject specific terms you come across, keeping a self-made “glossary of terms” can be extremely helpful at this point to help you access the content easier. You will hear multiple key terms mentioned in the tutorial playlist, please keep a note of these.</li> </ul>
<p><b>Overview of course</b></p>	<ul style="list-style-type: none"> <li>✓ <b>Exam Board</b> – AQA</li> <li>✓ <b>External exams</b> – available at end of Year 12 (AS Level) or Year 13 (full A Level)</li> <li>✓ <b>Themes, Exams, Assessment details and Timelines</b> – detailed below for Year 12 (AS work) and Year 13 (full A Level)</li> </ul>
<p><b>Short term focus: Paper 1 Programming</b></p>	<p>Paper 1 focuses primarily on <b>Problem Solving</b> and how this is portrayed through programming languages and algorithms. To make the most of your time before starting Year 12, I would like you to start learning the VB.NET programming language (not to be confused with VB6). You can find the “Visual Studio <b>Community</b>” software to install from this link <a href="https://visualstudio.microsoft.com/">https://visualstudio.microsoft.com/</a></p> <p>Please note that it is quite a large program and will take a lot of space. If you cannot install the program, then web versions are available but they are not as flexible <a href="https://dotnetfiddle.net/">https://dotnetfiddle.net/</a></p> <p>Follow the Visual Basic tutorial videos to practice coding in VB.NET (these use an older version of Visual Studio, but can still be followed make sure you have selected the Visual Basic language before clicking “Console Application” see below)</p>

## Create a new project

### Recent project templates

- Python Application Python
- Console App (.NET Framework) Visual Basic
- Windows Forms App (.NET Framework) Visual Basic
- Console App (.NET Core) C#

Search for project templates

Language Platform Project type

Filtering by: Visual Basic

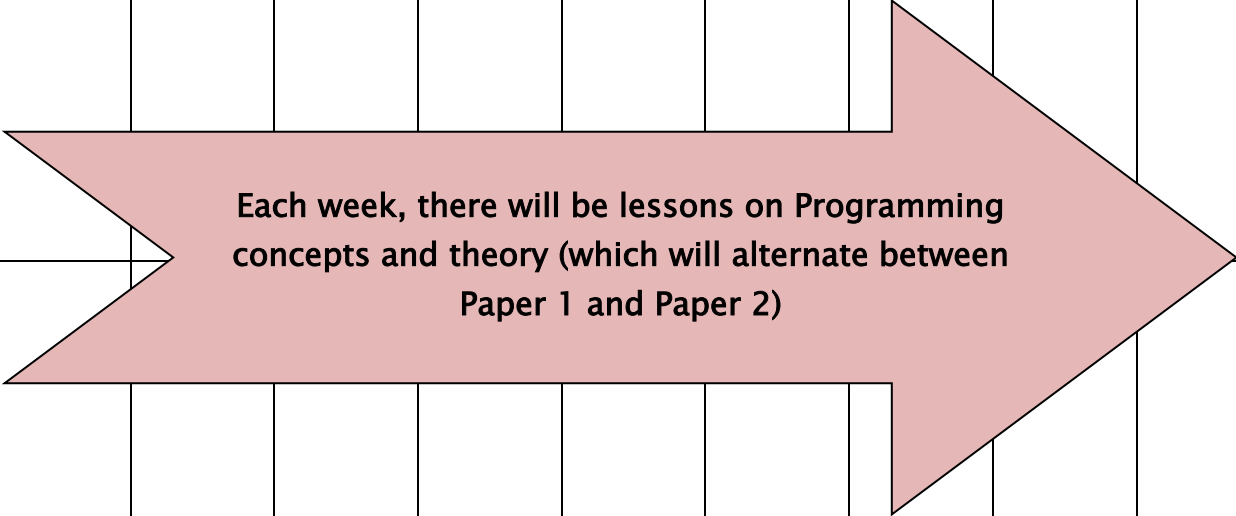
- Console App (.NET Framework) A project for creating a comm...  
Visual Basic Windows
- WPF App (.NET Framework) Windows Presentation Found...  
Visual Basic Windows Desktop
- Class Library (.NET Standard) A project for creating a class library that targets .NET Standard.

### Tutorial Playlist Link:

<https://www.youtube.com/watch?v=1mg3NHwGkas&list=PLC601DEA22187BBF1&index=4>

Good Luck! 😊

## AS Computer Science – Year 12

	September	October	November	December	January	February	March	April	May
<p><b>Paper 1: On-screen exam</b>                      (We study: Practical programming skills, with a heavy focus on Algorithms and Problem Solving)  <b>50%</b>                      1 hour and 45 minutes, 75 marks</p> <p>☐ Questions mainly range from 1 – 4 marks, and programming questions will range from 5 – 12 marks</p>	Late Sep Reflection	Mid-term coding project				Mid-term coding project			
<p><b>Paper 2: Written exam</b>                      (We study: Data Representation, Computer Systems, Computer Architecture, Ethics and Networking)  <b>50%</b>                      1 hour and 30 minutes, 75 marks</p> <p>☐ Questions mainly range from 1 – 4 marks, and often contains a longer written answer of 6 – 9 marks</p>	 <p>Each week, there will be lessons on Programming concepts and theory (which will alternate between Paper 1 and Paper 2)</p>								
				Mock Exam					Internal / External Exam

## Full A Level Computer Science – (builds on Year 12 work – additional non-exam project to complete) – Year 13

	September	October	November	December	January	February	March	April	May	June
<p><b>Paper 1: On-screen exam</b>                      (Assesses: all Y12 paper 1 topics, and goes into greater detail on algorithmic complexity)                      40% of A-level                      2 hours and 30 minutes, 100 marks</p> <p>□ Questions on this paper tend to range from 1–6 marks, as well as multiple larger programming questions which range from 8 – 12 marks.</p>	<p>Programming lessons to develop problem solving skills and also introduce Y13 techniques, including Object-Oriented techniques (ranges from 1-2 lessons a week)</p>									
	<p>Theory which recaps Y12 content, but also goes further to introduce Y13 topics for Paper 1 and Paper 2 (usually 2 lessons a week). Course specification</p>									
<p><b>Paper 2: Written exam</b>                      (Assesses: all Y12 paper 2 topics, and goes into greater detail on The Internet, Databases, Big Data, and Functional Programming)                      40% of A-level                      2 hours and 30 minutes, 100 marks</p>	<p>Time developing chosen coursework project (usually 2 lessons a week, until March)</p>									
								<p>Revision for exams</p>		
<p><b>NEA: The Programming Project</b>                      (Assesses: your programming skills when developing a much larger solution to a stakeholder’s problem)                      20% of A-level</p>										