

# OCR Computing J276 – Check List



Suggestions for resources that will help you revise are as below.

- Cambridge Elevate on line resource <https://elevate.cambridge.org/Elevate/Login.aspx#b> (All students can create free accounts on line)
  - GCSE OCR Computer Science (For the Grade -9-1 Course) The Revision Guide. (All students should have a copy)
  - GCSE OCR Computer Science (For the Grade -9-1 Course) Exam Practice Workbook– (All students should have a copy)
  - BBC Bitesize - <http://www.bbc.co.uk/education/subjects/z34k7ty>
1. **Written Paper 1 – 1.5 Hour exam - Content of Computer systems (J276/01) 50%**
  2. **Written Paper 2 – 1.5 Hour exam - Content of Computational thinking, algorithms and programming (J276/02) 50%**

<b>Written Paper 1 - Content of Computer systems (J276/01)</b>			
<b>1.1 Systems architecture</b>			
<ul style="list-style-type: none"> <li>• the purpose of the CPU</li> </ul>			
Von Neumann architecture: <ul style="list-style-type: none"> <li>• MAR (Memory Address Register)</li> <li>• MDR (Memory Data Register)</li> <li>• Program Counter</li> <li>• Accumulator</li> </ul>			
common CPU components and their function: <ul style="list-style-type: none"> <li>• ALU (Arithmetic Logic Unit)</li> <li>• CU (Control Unit)</li> <li>• Cache</li> </ul>			
the function of the CPU as fetch and execute instructions stored in memory			
how common characteristics of CPUs affect their performance:			
<ul style="list-style-type: none"> <li>• clock speed</li> <li>• cache size</li> <li>• number of cores</li> </ul>			
embedded systems:			
<ul style="list-style-type: none"> <li>• purpose of embedded systems</li> <li>• examples of embedded systems.</li> </ul>			
<b>1.2 Memory</b>			
<ul style="list-style-type: none"> <li>• the difference between RAM and ROM</li> <li>• the purpose of ROM in a computer system</li> <li>• the purpose of RAM in a computer system</li> <li>• the need for virtual memory</li> <li>• Flash memory</li> </ul>			
<b>1.3 Storage</b>			
<ul style="list-style-type: none"> <li>• the need for secondary storage</li> <li>• data capacity and calculation of data capacity requirements</li> </ul>			

common types of storage:			
<ul style="list-style-type: none"> <li>• optical</li> <li>• magnetic</li> <li>• solid state</li> </ul>			

suitable storage devices and storage media for a given application, and the advantages and disadvantages of these, using characteristics:			
<ul style="list-style-type: none"> <li>• capacity</li> </ul>			
<ul style="list-style-type: none"> <li>• speed</li> </ul>			
<ul style="list-style-type: none"> <li>• portability</li> </ul>			
<ul style="list-style-type: none"> <li>• durability</li> </ul>			
<ul style="list-style-type: none"> <li>• reliability</li> </ul>			
<ul style="list-style-type: none"> <li>• Cost.</li> </ul>			

## 1.4 Wired and wireless networks

types of networks:	😊	😐	😞
<ul style="list-style-type: none"> <li>• LAN (Local Area Network)</li> </ul>			
<ul style="list-style-type: none"> <li>• WAN (Wide Area Network)</li> </ul>			
<ul style="list-style-type: none"> <li>• factors that affect the performance of networks</li> </ul>			
<ul style="list-style-type: none"> <li>• the different roles of computers in a client-server and a peer-to-peer network</li> </ul>			
the hardware needed to connect stand-alone computers into a Local Area Network:			
<ul style="list-style-type: none"> <li>• wireless access points</li> </ul>			
<ul style="list-style-type: none"> <li>• routers/switches</li> </ul>			
<ul style="list-style-type: none"> <li>• NIC (Network Interface Controller/Card)</li> </ul>			
<ul style="list-style-type: none"> <li>• transmission media</li> </ul>			
the internet as a worldwide collection of computer networks:			
<ul style="list-style-type: none"> <li>• DNS (Domain Name Server)</li> </ul>			
<ul style="list-style-type: none"> <li>• hosting</li> </ul>			
<ul style="list-style-type: none"> <li>• the cloud</li> </ul>			
<ul style="list-style-type: none"> <li>• the concept of virtual networks.</li> </ul>			

## 1.5 Network topologies, protocols and layers

<ul style="list-style-type: none"> <li>• star and mesh network topologies</li> </ul>	😊	😐	😞
Wifi:			
<ul style="list-style-type: none"> <li>• frequency and channels</li> <li>• encryption</li> <li>• ethernet</li> </ul>			
the uses of IP addressing, MAC addressing, and protocols including:			
<ul style="list-style-type: none"> <li>• TCP/IP (Transmission Control Protocol/Internet Protocol)</li> </ul>			
<ul style="list-style-type: none"> <li>• HTTP (Hyper Text Transfer Protocol)</li> </ul>			
<ul style="list-style-type: none"> <li>• HTTPS (Hyper Text Transfer Protocol Secure)</li> </ul>			
<ul style="list-style-type: none"> <li>• FTP (File Transfer Protocol)</li> </ul>			
<ul style="list-style-type: none"> <li>• POP (Post Office Protocol)</li> </ul>			
<ul style="list-style-type: none"> <li>• IMAP (Internet Message Access Protocol)</li> </ul>			
<ul style="list-style-type: none"> <li>• SMTP (Simple Mail Transfer Protocol)</li> </ul>			

<ul style="list-style-type: none"> <li>the concept of layers</li> </ul>			
<ul style="list-style-type: none"> <li>packet switching.</li> </ul>			
<b>1.6 System security</b>	😊	😐	😞
<ul style="list-style-type: none"> <li>forms of attack</li> </ul>			
threats posed to networks:			
<ul style="list-style-type: none"> <li>malware</li> </ul>			
<ul style="list-style-type: none"> <li>phishing</li> </ul>			
<ul style="list-style-type: none"> <li>people as the 'weak point' in secure systems (social engineering)</li> </ul>			
<ul style="list-style-type: none"> <li>brute force attacks</li> </ul>			
<ul style="list-style-type: none"> <li>denial of service attacks</li> </ul>			
<ul style="list-style-type: none"> <li>data interception and theft</li> </ul>			
<ul style="list-style-type: none"> <li>the concept of SQL injection</li> </ul>			
<ul style="list-style-type: none"> <li>poor network policy</li> </ul>			
<ul style="list-style-type: none"> <li>malware</li> </ul>			
identifying and preventing vulnerabilities:			
<ul style="list-style-type: none"> <li>penetration testing</li> </ul>			
<ul style="list-style-type: none"> <li>network forensics</li> </ul>			
<ul style="list-style-type: none"> <li>network policies</li> </ul>			
<ul style="list-style-type: none"> <li>anti-malware software</li> </ul>			
<ul style="list-style-type: none"> <li>firewalls</li> </ul>			
<ul style="list-style-type: none"> <li>user access levels</li> </ul>			
<ul style="list-style-type: none"> <li>passwords</li> </ul>			
<ul style="list-style-type: none"> <li>encryption</li> </ul>			
<b>1.7 Systems software</b>			
	😊	😐	😞
the purpose and functionality of systems software			
operating systems:			
<ul style="list-style-type: none"> <li>user interface</li> </ul>			
<ul style="list-style-type: none"> <li>memory management/multitasking</li> </ul>			
<ul style="list-style-type: none"> <li>peripheral management and drivers</li> </ul>			
<ul style="list-style-type: none"> <li>user management</li> </ul>			
<ul style="list-style-type: none"> <li>file management</li> </ul>			
<ul style="list-style-type: none"> <li>user interface</li> </ul>			
utility system software:			
<ul style="list-style-type: none"> <li>encryption software</li> </ul>			
<ul style="list-style-type: none"> <li>defragmentation</li> </ul>			
<ul style="list-style-type: none"> <li>data compression</li> </ul>			
<ul style="list-style-type: none"> <li>the role and methods of backup</li> </ul>			
<b>1.8 Ethical, legal, cultural and environmental concerns</b>	😊	😐	😞
how to investigate and discuss Computer Science technologies while considering:			
<ul style="list-style-type: none"> <li>ethical issues</li> </ul>			
<ul style="list-style-type: none"> <li>legal issues</li> </ul>			
<ul style="list-style-type: none"> <li>cultural issues</li> </ul>			
<ul style="list-style-type: none"> <li>environmental issues</li> </ul>			
<ul style="list-style-type: none"> <li>privacy issues</li> </ul>			

<ul style="list-style-type: none"> <li>• how key stakeholders are affected by technologies</li> </ul>			
<ul style="list-style-type: none"> <li>• environmental impact of Computer Science</li> </ul>			
<ul style="list-style-type: none"> <li>• cultural implications of Computer Science</li> </ul>			
<ul style="list-style-type: none"> <li>• open source vs proprietary software</li> </ul>			
legislation relevant to Computer Science:			
<ul style="list-style-type: none"> <li>• The Data Protection Act 1998</li> </ul>			
<ul style="list-style-type: none"> <li>• Computer Misuse Act 1990</li> </ul>			
<ul style="list-style-type: none"> <li>• Copyright Designs and Patents Act 1988</li> </ul>			
<ul style="list-style-type: none"> <li>• Creative Commons Licensing</li> </ul>			
<ul style="list-style-type: none"> <li>• Freedom of Information Act 2000.</li> </ul>			
Written Paper 2 - Content of Computational thinking, algorithms and programming (J276/02)	☺	☹	☹
<b>2.1 Algorithms</b>			
computational thinking:			
<ul style="list-style-type: none"> <li>• abstraction</li> </ul>			
<ul style="list-style-type: none"> <li>• decomposition</li> </ul>			
<ul style="list-style-type: none"> <li>• algorithmic thinking</li> </ul>			
standard searching algorithms:			
<ul style="list-style-type: none"> <li>• binary search</li> </ul>			
<ul style="list-style-type: none"> <li>• linear search</li> </ul>			
standard sorting algorithms:			
<ul style="list-style-type: none"> <li>• bubble sort</li> </ul>			
<ul style="list-style-type: none"> <li>• merge sort</li> </ul>			
<ul style="list-style-type: none"> <li>• insertion sort</li> </ul>			
how to produce algorithms using:			
<ul style="list-style-type: none"> <li>• pseudocode</li> </ul>			
<ul style="list-style-type: none"> <li>• using flow diagrams</li> </ul>			
<ul style="list-style-type: none"> <li>• interpret, correct or complete algorithms</li> </ul>			
<b>2.2 Programming techniques</b>	☺	☹	☹
<ul style="list-style-type: none"> <li>• the use of variables, constants, operators, inputs, outputs and assignments</li> </ul>			
the use of the three basic programming constructs used to control the flow of a program:			
<ul style="list-style-type: none"> <li>• sequence</li> </ul>			
<ul style="list-style-type: none"> <li>• selection</li> </ul>			
<ul style="list-style-type: none"> <li>• iteration (count and condition controlled loops)</li> </ul>			
<ul style="list-style-type: none"> <li>• the use of basic string manipulation</li> </ul>			
the use of basic file handling operations:			
<ul style="list-style-type: none"> <li>• open</li> </ul>			
<ul style="list-style-type: none"> <li>• read</li> </ul>			
<ul style="list-style-type: none"> <li>• write</li> </ul>			
<ul style="list-style-type: none"> <li>• close</li> </ul>			
<ul style="list-style-type: none"> <li>• the use of records to store data</li> </ul>			
<ul style="list-style-type: none"> <li>• the use of SQL to search for data</li> </ul>			
<ul style="list-style-type: none"> <li>• the use of arrays (or equivalent) when solving problems, including both one and two dimensional</li> </ul>			
<ul style="list-style-type: none"> <li>• arrays</li> </ul>			
<ul style="list-style-type: none"> <li>• how to use sub programs (functions and procedures) to produce structured code</li> </ul>			
the use of data types:			
<ul style="list-style-type: none"> <li>• integer</li> </ul>			
<ul style="list-style-type: none"> <li>• real</li> </ul>			
<ul style="list-style-type: none"> <li>• Boolean</li> </ul>			
<ul style="list-style-type: none"> <li>• character and string</li> </ul>			

<ul style="list-style-type: none"> <li>• casting</li> </ul>			
<ul style="list-style-type: none"> <li>• the common arithmetic operators</li> </ul>			
<ul style="list-style-type: none"> <li>• the common Boolean operators.</li> </ul>			
<b>2.3 Producing robust programs</b>	😊	😐	😞
defensive design considerations:			
<ul style="list-style-type: none"> <li>• input sanitisation/validation</li> </ul>			
<ul style="list-style-type: none"> <li>• planning for contingencies</li> </ul>			
<ul style="list-style-type: none"> <li>• anticipating misuse</li> </ul>			
<ul style="list-style-type: none"> <li>• authentication</li> </ul>			
maintainability:			
<ul style="list-style-type: none"> <li>• comments</li> </ul>			
<ul style="list-style-type: none"> <li>• indentation</li> </ul>			
<ul style="list-style-type: none"> <li>• the purpose of testing</li> </ul>			
types of testing:			
<ul style="list-style-type: none"> <li>• iterative</li> </ul>			
<ul style="list-style-type: none"> <li>• final/terminal</li> </ul>			
<ul style="list-style-type: none"> <li>• how to identify syntax and logic errors</li> </ul>			
<ul style="list-style-type: none"> <li>• selecting and using suitable test data</li> </ul>			
<b>2.4 Computational logic</b>	😊	😐	😞
<ul style="list-style-type: none"> <li>• why data is represented in computer systems in binary form</li> </ul>			
<ul style="list-style-type: none"> <li>• simple logic diagrams using the operations AND, OR and NOT</li> </ul>			
<ul style="list-style-type: none"> <li>• truth tables</li> </ul>			
<ul style="list-style-type: none"> <li>• combining Boolean operators using AND, OR and NOT to two levels</li> </ul>			
<ul style="list-style-type: none"> <li>• applying logical operators in appropriate truth tables to solve problems</li> </ul>			
applying computing-related mathematics:			
<ul style="list-style-type: none"> <li>• +</li> </ul>			
<ul style="list-style-type: none"> <li>• –</li> </ul>			
<ul style="list-style-type: none"> <li>• /</li> </ul>			
<ul style="list-style-type: none"> <li>• *</li> </ul>			
<ul style="list-style-type: none"> <li>• Exponentiation (^)</li> </ul>			
<ul style="list-style-type: none"> <li>• MOD</li> </ul>			
<ul style="list-style-type: none"> <li>• DIV</li> </ul>			
<b>2.5 Translators and facilities of languages</b>	😊	😐	😞
<ul style="list-style-type: none"> <li>• characteristics and purpose of different levels of programming language, including low level languages</li> </ul>			
<ul style="list-style-type: none"> <li>• the purpose of translators</li> </ul>			
<ul style="list-style-type: none"> <li>• the characteristics of an assembler, a compiler and an interpreter</li> </ul>			
common tools and facilities available in an integrated development environment (IDE):			
<ul style="list-style-type: none"> <li>• editors</li> </ul>			
<ul style="list-style-type: none"> <li>• error diagnostics</li> </ul>			
<ul style="list-style-type: none"> <li>• run-time environment</li> </ul>			
<ul style="list-style-type: none"> <li>• translators.</li> </ul>			
<b>2.6 Data representation</b>	😊	😐	😞
unit			
<ul style="list-style-type: none"> <li>• bit, nibble, byte, kilobyte, megabyte, gigabyte, terabyte, petabyte</li> </ul>			
<ul style="list-style-type: none"> <li>• how data needs to be converted into a binary format to be processed by a computer.</li> </ul>			
Numbers			
<ul style="list-style-type: none"> <li>• how to convert positive denary whole numbers (0–255) into 8 bit binary numbers and vice versa</li> </ul>			

<ul style="list-style-type: none"> <li>• how to add two 8 bit binary integers and explain overflow errors which may occur</li> </ul>			
<ul style="list-style-type: none"> <li>• binary shifts</li> </ul>			
<ul style="list-style-type: none"> <li>• how to convert positive denary whole numbers (0–255) into 2 digit hexadecimal numbers and vice versa</li> </ul>			
<ul style="list-style-type: none"> <li>• how to convert from binary to hexadecimal equivalents and vice versa</li> </ul>			
<ul style="list-style-type: none"> <li>• check digits.</li> </ul>			
<ul style="list-style-type: none"> <li>• Characters</li> </ul>			
<ul style="list-style-type: none"> <li>• the use of binary codes to represent characters</li> </ul>			
<ul style="list-style-type: none"> <li>• the term 'character-set'</li> </ul>			
<ul style="list-style-type: none"> <li>• the relationship between the number of bits per character in a character set and the number of</li> </ul>			
<ul style="list-style-type: none"> <li>• characters which can be represented (for example ASCII, extended ASCII and Unicode).</li> </ul>			
<ul style="list-style-type: none"> <li>• Images</li> </ul>			
<ul style="list-style-type: none"> <li>• how an image is represented as a series of pixels represented in binary</li> </ul>			
<ul style="list-style-type: none"> <li>• metadata included in the file</li> </ul>			
<ul style="list-style-type: none"> <li>• the effect of colour depth and resolution on the size of an image file.</li> </ul>			
<ul style="list-style-type: none"> <li>• Sound</li> </ul>			
<ul style="list-style-type: none"> <li>• how sound can be sampled and stored in digital form</li> </ul>			
how sampling intervals and other factors affect the size of a sound file and the quality of its playback:			
<ul style="list-style-type: none"> <li>• sample size</li> </ul>			
<ul style="list-style-type: none"> <li>• bit rate</li> </ul>			
<ul style="list-style-type: none"> <li>• sampling frequency.</li> </ul>			
<ul style="list-style-type: none"> <li>• Compression</li> </ul>			
<ul style="list-style-type: none"> <li>• need for compression</li> </ul>			
types of compression:			
<ul style="list-style-type: none"> <li>• lossy</li> </ul>			
<ul style="list-style-type: none"> <li>• lossless</li> </ul>			