Cambridge
O Level

Cambridge Assessment International Education
Cambridge Ordinary Level

COMPUTER SCIENCE **2210/13**

Paper 1 **October/November 2019**

MARK SCHEME

Maximum Mark: 75

Published

This mark scheme is published as an aid to teachers and candidates, to indicate the requirements of the examination. It shows the basis on which Examiners were instructed to award marks. It does not indicate the details of the discussions that took place at an Examiners' meeting before marking began, which would have considered the acceptability of alternative answers.

Mark schemes should be read in conjunction with the question paper and the Principal Examiner Report for Teachers.

Cambridge International will not enter into discussions about these mark schemes.

Cambridge International is publishing the mark schemes for the October/November 2019 series for most Cambridge IGCSE™, Cambridge International A and AS Level components and some Cambridge O Level components.

Generic Marking Principles

These general marking principles must be applied by all examiners when marking candidate answers. They should be applied alongside the specific content of the mark scheme or generic level descriptors for a question. Each question paper and mark scheme will also comply with these marking principles.

GENERIC MARKING PRINCIPLE 1:

Marks must be awarded in line with:

- the specific content of the mark scheme or the generic level descriptors for the question
- the specific skills defined in the mark scheme or in the generic level descriptors for the question
- the standard of response required by a candidate as exemplified by the standardisation scripts.

GENERIC MARKING PRINCIPLE 2:

Marks awarded are always **whole marks** (not half marks, or other fractions).

GENERIC MARKING PRINCIPLE 3:

Marks must be awarded **positively**:

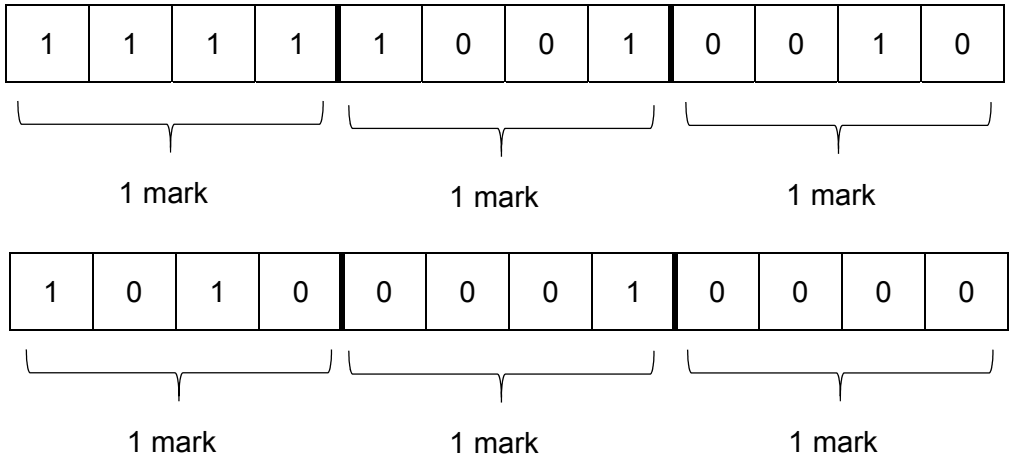
- marks are awarded for correct/valid answers, as defined in the mark scheme. However, credit is given for valid answers which go beyond the scope of the syllabus and mark scheme, referring to your Team Leader as appropriate
- marks are awarded when candidates clearly demonstrate what they know and can do
- marks are not deducted for errors
- marks are not deducted for omissions
- answers should only be judged on the quality of spelling, punctuation and grammar when these features are specifically assessed by the question as indicated by the mark scheme. The meaning, however, should be unambiguous.

GENERIC MARKING PRINCIPLE 4:

Rules must be applied consistently e.g. in situations where candidates have not followed instructions or in the application of generic level descriptors.

GENERIC MARKING PRINCIPLE 5: Marks should be awarded using the full range of marks defined in the mark scheme for the question (however; the use of the full mark range may be limited according to the quality of the candidate responses seen).
GENERIC MARKING PRINCIPLE 6: Marks awarded are based solely on the requirements as defined in the mark scheme. Marks should not be awarded with grade thresholds or grade descriptors in mind.

Question	Answer	Marks
1(a)(i)	**Two** from: 2D scanner Touchscreen Keypad/keyboard Card reader Mouse Digital camera	**2**
1(a)(ii)	**Two** from: HDD SSD USB flash memory drive SD card Any optical	**2**
1(a)(iii)	**Two** from: Monitor/Touch screen Speaker Printer LED // Light	**2**
1(b)(i)	Increase the length of the key // make key 12-bit, etc.	**1**
1(b)(ii)	Cypher text	**1**

Question	Answer	Marks
1(b)(iii)	**Six** from: The system could use odd or even parity A parity bit is added The data is checked to see if it has incorrect/correct parity // by example If parity is correct no error is found An acknowledgement is sent that data is received correctly The next packet of data is transmitted If incorrect parity is found an error has occurred A signal is sent back to request the data is resent The data is resent until data is received correctly/timeout occurs	**6**
1(c)(i)		**6**

Question	Answer	Marks
1(c)(ii)	**One** mark for identification: Compression **Three** from e.g.: Best compression would be lossy Use compression algorithm This would remove all the unnecessary data from the file // removes detail/sound that the human eye/ear may not see/hear Reduce colour palette … … so each pixel requires fewer bits Reduce resolution Only store what changes between frames // temporal redundancy	**4**
1(d)	**Five** from: The display is made up of pixels … … that are arranged together as a matrix Each pixel has three filters, red, blue and green Shades of colour are achieved by mixing red, blue and green The screen is backlit Light is shone through the liquid crystals The liquid crystals can be made to turn solid or transparent/on or off … … by changing the shape of the crystal	**5**

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>2(a)</td><td>One mark for each correct row
<table>
<tr><th>Statement</th><th>True (✓)</th><th>False (✓)</th></tr>
<tr><td>High-level languages need to be translated into machine code to run on a computer</td><td>✓</td><td></td></tr>
<tr><td>High-level languages are written using mnemonic codes</td><td></td><td>✓</td></tr>
<tr><td>High-level languages are specific to the computer’s hardware</td><td></td><td>✓</td></tr>
<tr><td>High-level languages are portable languages</td><td>✓</td><td></td></tr>
</table></td><td>4</td></tr>
</table>

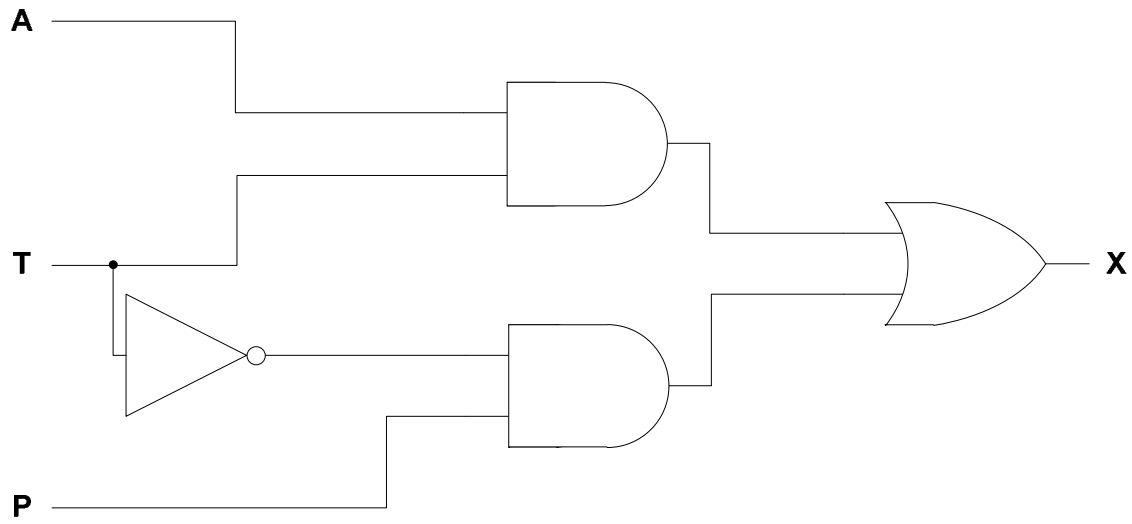
Question	Answer	Marks
2(b)	**One** mark for the correct tick	**1**

Example program	Tick (✓)
`1011100000110000` `0000011011100010`	

`INP`
`STA ONE`
`INP`
`STA TWO`
`ADD ONE`	

`a = input()`
`b = input()`
`if a == b:`
`  print("Correct")`
`else:`
`  print("Incorrect")`
	✓

Question	Answer	Marks
3	**One** mark for each correct term in the correct order Serial Parallel Serial Simplex Parallel	**5**

Question	Answer	Marks
4(a)	**One** mark for each correct logic gate with correct input(s)  	**4**

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>4(b)</td><td>Four mark for 8 correct outputs
Three marks for 6 or 7 correct outputs
Two mark for 4 or 5 correct outputs
One mark for 2 or 3 correct outputs

<table>
<tr><th>A</th><th>T</th><th>P</th><th>Working space</th><th>X</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td></td><td>0</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td></td><td>1</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td></td><td>0</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td></td><td>0</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td></td><td>0</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td></td><td>1</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td></td><td>1</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td></td><td>1</td></tr>
</table></td><td>4</td></tr>
<tr><td>4(c)</td><td>Six from:
Sensor sends a signal/reading/data to the microprocessor
Signal/reading/data is analogue and is converted to digital using ADC
Reading/data is stored in the system
Microprocessor compares data/reading to the pre-set value of 7
If value is greater than 7 …
… a signal/data is sent by the microprocessor to display a warning message on a monitor
The process is continuous</td><td>6</td></tr>
</table>

Question	Answer	Marks
5	**One** mark for each correct parity bit	**3**

	Parity bit							
Register A	**0**	0	1	0	0	0	1	1
Register B	**0**	0	0	0	0	1	1	1
Register C	**0**	0	0	0	0	0	1	1

Question	Answer	Marks
6(a)	Free software	**1**
6(b)	Freeware	**1**
6(c)	Shareware	**1**
6(d)	Plagiarism // Intellectual property theft	**1**
6(e)	Copyright	**1**

Question	Answer	Marks
7(a)(i)	**Three** from: RAM Primary memory Volatile memory Holds currently in use data/instructions Directly accessed by the CPU	**3**
7(a)(ii)	**Two** from: Arithmetic and logic unit (ALU) Memory address register (MAR) Memory data register (MDR) // Memory buffer register (MBR) Accumulator (ACC) Immediate Access Store (IAS) Control Unit (CU) Program counter (PC) Current instruction register (CIR) Address bus Data bus Control bus Input device Output device Secondary storage device	**2**

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>7(b)</td><td>One mark for each correct row

<table>
<tr><th>Statement</th><th>True (✓)</th><th>False (✓)</th></tr>
<tr><td>Interrupts can be hardware based or software based</td><td>✓</td><td></td></tr>
<tr><td>Interrupts are handled by the operating system</td><td>✓</td><td></td></tr>
<tr><td>Interrupts allow a computer to multitask</td><td>✓</td><td></td></tr>
<tr><td>Interrupts work out which program to give priority to</td><td></td><td>✓</td></tr>
<tr><td>Interrupts are vital to a computer and it cannot function without them</td><td>✓</td><td></td></tr>
</table></td><td>5</td></tr>
</table>

Question	Answer	Marks
8	**Four** from: A hacker could have hacked the network … … and downloaded the malware onto the network Clicking a link/attachment/downloaded a file from an email/on a webpage … … the malware could have been embedded into the link/attachment/file Opening an infected software package … … this would trigger the malware to download onto the network Inserting an infected portable storage device … … when the drive is accessed the malware is downloaded to the network Firewall has been turned off … … so malware would not be detected/checked for when entering network Anti-malware has been turned off … … so malware is not detected/checked for when files are downloaded	**4**