Cambridge Assessment
International Education

Grade thresholds – November 2017

Cambridge IGCSE Computer Science (0478)

Grade thresholds taken for Syllabus 0478 (Computer Science) in the November 2017 examination.

		minimum raw mark required for grade:						
	maximum raw mark available	A	B	C	D	E	F	G
Component 11	75	48	40	33	28	23	18	13
Component 12	75	49	41	34	28	23	18	13
Component 13	75	48	40	33	28	23	18	13
Component 21	50	31	23	16	12	9	6	3
Component 22	50	34	26	19	15	11	7	3
Component 23	50	31	23	16	12	9	6	3

Grade A* does not exist at the level of an individual component.

The maximum total mark for this syllabus, after weighting has been applied, is **125**.

The overall thresholds for the different grades were set as follows.

Option	Combination of Components	A*	A	B	C	D	E	F	G
AX	11, 21	93	78	63	49	40	32	24	16
AY	12, 22	97	82	67	53	43	34	25	16
AZ	13, 23	93	78	63	49	40	32	24	16

Cambridge Assessment International Education
Cambridge International General Certificate of Secondary Education

---

COMPUTER SCIENCE **0478/11**
Paper 1 **October/November 2017**
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 2017 series for most Cambridge IGCSE®, Cambridge International A and AS Level components and some Cambridge O Level components.

This syllabus is approved for use in England, Wales and Northern Ireland as a Cambridge International Level 1/Level 2 Certificate.

This document consists of **10** printed pages.

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>1(a)</td><td>Output</td><td>1</td></tr>
<tr><td>1(b)</td><td>1 mark for each correct conversion
<table>
<tr><td colspan="4"><i>E</i></td><td colspan="4"><i>0</i></td><td colspan="4"><i>4</i></td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td></tr>
</table></td><td>3</td></tr>
<tr><td>1(c)</td><td>Any one from:
– Hexadecimal codes can fit in a smaller display rather than a full text based message
– Smaller amount of memory needed to store the hex error messages than text based</td><td>1</td></tr>
<tr><td>1(d)</td><td>1 mark for correct sensor, 1 mark for corresponding use
Possible examples could include:

– Temperature (sensor)
– To monitor the temperature of the water

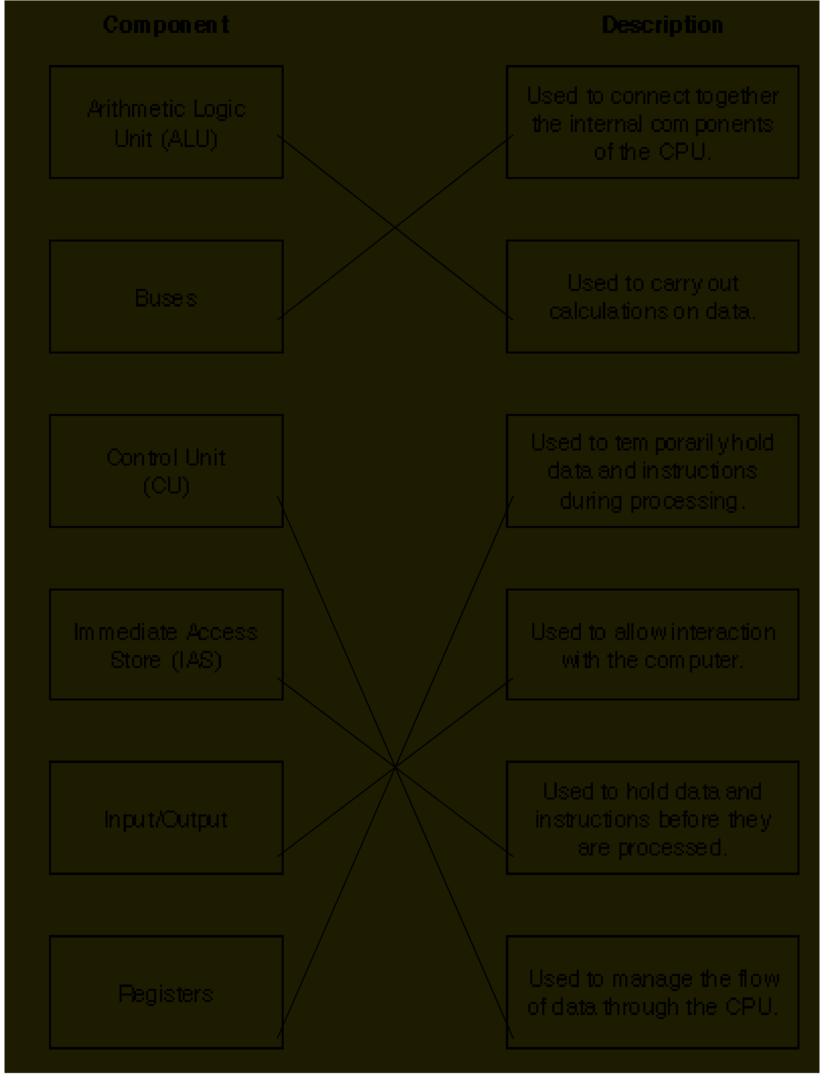
– Pressure (sensor)
– To monitor the level of water in the washing machine

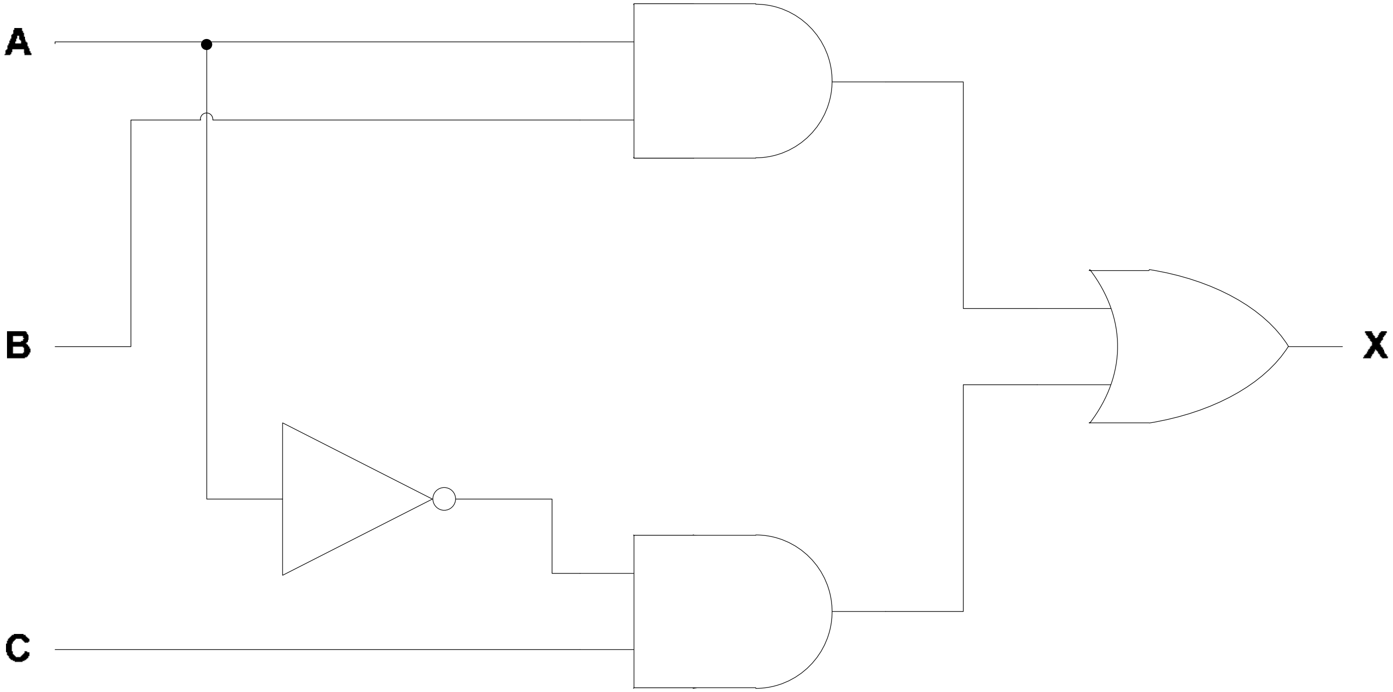
– Motion (sensor)
– To monitor whether the drum is still in motion

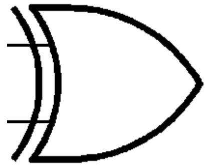
– pH (sensor)
– To monitor the level of water hardness/detergent present in the water</td><td>6</td></tr>
</table>

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>2</td><td>1 mark for each correct file format e.g.
<table>
<tr><th>File type</th><th>File format</th></tr>
<tr><td>Pictures</td><td>.JPEG</td></tr>
<tr><td>Text</td><td>.doc, .txt, .rtf,
.docx, .odt .pdf</td></tr>
<tr><td>Sound</td><td>.mp3, .wav, .aif,
.flac, .mid</td></tr>
<tr><td>Video</td><td>.mp4, .flv, .wmv</td></tr>
</table></td><td>3</td></tr>
</table>

Question	Answer	Marks
3(a)	– Part 1 (access) protocol – Part 2 domain (name) – Part 3 filename	**3**
3(b)	**Four** from: – IP address is used to identify a device (on the Internet / network) – IP address is allocated by the network/ ISP – Can be used in place of URL – IP addresses can be IPv4 or IPv6 – IP address can be static … – ... meaning it doesn't change each time it is connected to the Internet – IP address can be dynamic …. – … meaning that it can change each time a device is connected to the Internet – Any valid example (e.g. xxx.xxx.xxx.xxx or xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx)	**4**

Question	Answer	Marks
4	1 mark for each correct line up to a total of 5 marks  	**5**

Question	Answer	Marks
5(a)	1 mark for each correct logic gate 	**4**

Question	Answer	Marks
5(b)	1 mark for correct logic gate symbol:  Any **four** from: – similar to an OR gate – It has (at least) two inputs – Output will be high/1 if both inputs are different – Output will be high/1 if either input is high – Output will be low/0 if both inputs are high – Output will be low/0 if both inputs are low	**5**

Question	Answer	Marks
6	Any **six** from: 2D – (Scanner) shines a light onto the surface of a document // Light moves across document – Reflected light is captured – Uses mirrors and lenses – Captured image is converted into a digital file – Produces a 2D digital image 3D – Scanners shines a laser (or light) over the surface of a 3D object – Records measurements of the geometry/dimensions of the object – Measurements are converted to digital file – Produces a 3D digital model	**6**

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>7</td><td>1 mark for each correct tick
<table>
<tr><th>Statement</th><th>true
(✓)</th><th>false
(✓)</th></tr>
<tr><td>Firewalls can monitor incoming and outgoing traffic.</td><td>✓</td><td></td></tr>
<tr><td>Firewalls operate by checking traffic against a set of rules.</td><td>✓</td><td></td></tr>
<tr><td>Firewalls cannot block access to a certain website.</td><td></td><td>✓</td></tr>
<tr><td>Firewalls can be software and hardware.</td><td>✓</td><td></td></tr>
<tr><td>Firewalls can act as intermediary servers.</td><td></td><td>✓</td></tr>
<tr><td>Firewalls can block unauthorised traffic.</td><td>✓</td><td></td></tr>
</table>
</td><td>6</td></tr>
</table>

Question	Answer	Marks
8(a)	Any **three** from: – Human error (e.g. deleting/overwriting data) – Physical damage – Power failure/surge – Hardware failure – Software crashing	**3**
8(b)	Any **three** from: – Online shopping // Online payment systems // Online booking – Email – Cloud based storage – Intranet/extranet – VPN – VoIP // video conferencing – Instant messaging (IM) // social networking // online gaming	**3**

Question	Answer	Marks
8(c)	1 mark for identifying, 1 mark for description – Strong password – To make it difficult to hack an account – Biometric device – To use data that is difficult to fake as a password – TLS // Encryption – To make data meaningless if intercepted – To encrypt data that is exchanged (TLS only) – More secure than SSL (TLS only) – Anti-spyware (software) – To find and remove any spyware that is installed on a computer – To help stop key loggers recording key presses – Firewall – To help prevent unauthorised access to an account – Blocks any requests that do not meet/match the criteria – Authentication (card reader at home)/mobile security code app/two-step verification – To add another level of identification of the user – Use of drop-down boxes (or equivalent) – So key loggers cannot record the key presses – Proxy server – To divert an attack away from the main system	**6**

Question	Answer	Marks
9(a)	Any **four** from: – (Red) laser is used – (Laser beams) shines onto surface of the disk – It is rotated (at a constant speed) to be read – Surface is covered in a track (that spirals from the centre) – Data is represented on the surface using pits and lands – Pits and lands represent binary values – Pits reflect light back differently (to the area in between/land) – Optical device can determine the binary value from the light reflection	**4**
9(b)	1 mark for calculation, 1 mark for correct answer: – 1000 × 16 – 16000/8 – Answer is **2000** bytes	**2**
9(c)	**Four** from: (Max 2 for either primary or secondary) – Primary RAM and ROM – Secondary HDD and SSD – Primary is directly accessible by CPU – Secondary is not directly accessible by CPU – Primary is internal to computer – Secondary can be internal **or** external to the computer – Primary stores boot up instructions **and** can hold data whilst being processed – Secondary stores files/software – Primary has faster access speed – Secondary has a slower access speed – Primary has both volatile **and** non-volatile – Secondary is non-volatile	**4**

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>10</td><td>1 mark for each correct tick<table>
<tr><th>Statement</th><th>true (✓)</th><th>false (✓)</th></tr>
<tr><td>Assembly language uses mnemonic codes.</td><td>✓</td><td></td></tr>
<tr><td>Assembly language programs do not need a translator to be executed.</td><td></td><td>✓</td></tr>
<tr><td>Assembly language is a low-level programming language.</td><td>✓</td><td></td></tr>
<tr><td>Assembly language is specific to the computer hardware.</td><td>✓</td><td></td></tr>
<tr><td>Assembly language is machine code.</td><td></td><td>✓</td></tr>
<tr><td>Assembly language is often used to create drivers for hardware.</td><td>✓</td><td></td></tr>
</table></td><td>6</td></tr>
</table>

Cambridge Assessment International Education
Cambridge International General Certificate of Secondary Education

---

COMPUTER SCIENCE **0478/12**

Paper 1 **October/November 2017**

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 2017 series for most Cambridge IGCSE®, Cambridge International A and AS Level components and some Cambridge O Level components.

This syllabus is approved for use in England, Wales and Northern Ireland as a Cambridge International Level 1/Level 2 Certificate.

This document consists of **9** printed pages.

Question	Answer	Marks
1	1 mark per correct instruction: 9 – LEFT 1 – DOWN C – OPEN 3 – CLOSE F – UP	**5**

Question	Answer	Marks
2	1 mark for each correct category: *HDD* – Secondary *RAM* – Primary *ROM* – Primary *CD-ROM* – Off-line *SSD* – Secondary *DVD-RAM* – Off-line	**6**

Question	Answer	Marks
3(a)	Any **four** from (**Max 2** per number system) : • A binary number system is a base-2 system • A denary number system is a base-10 system • A binary number system uses 0 and 1 values • A denary number system uses 0 to 9 values • A binary number system has units/ placeholders/column headings that increase by the power of 2 • A denary number system has units/ placeholders/column headings that increase by the power of 10 • Binary has more digit for the same value// Denary has less digits for the same value	**4**

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>3(b)</td><td>Five from:
• Correct column headings / place holders by example
• Correctly place a 1 or a 0 for each column
• Identify the columns to be added
• Add together the (denary) values identified …
• … this will give a total which is the denary number/answer
• Answer is 10</td><td>5</td></tr>
</table>

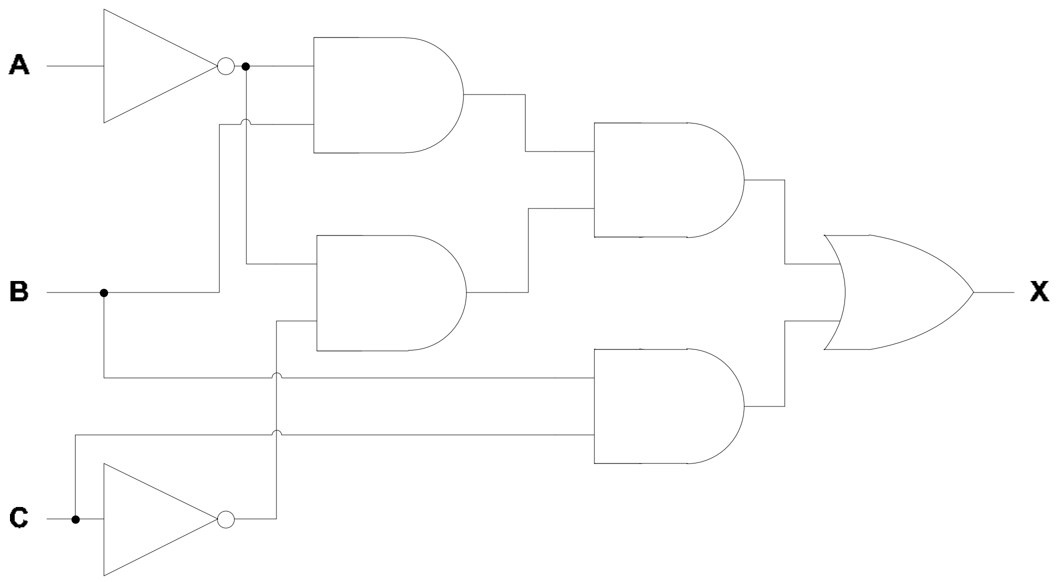
<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>4(a)(i)</td><td>
<table>
<tr><th>Method 1</th><th>Tick (✓)</th><th>Method 2</th><th>Tick (✓)</th></tr>
<tr><td>Serial</td><td>✓</td><td>Simplex</td><td></td></tr>
<tr><td>Parallel</td><td></td><td>Half-duplex</td><td></td></tr>
<tr><td></td><td></td><td>Duplex</td><td>✓</td></tr>
</table>
</td><td>2</td></tr>
<tr><td>4(a)(ii)</td><td>Any four from (Max 3 for serial):
• Serial has less/lower interference
• Serial is (more) reliable/accurate over distances
• In serial the bits won’t be skewed
• In serial it is easier to collate the bits together again after transmission

• Duplex transmits data in both directions at the same time
• simplex/half-duplex/remaining methods won’t allow read and write at same time</td><td>4</td></tr>
</table>

Question	Answer	Marks
4(b)	1 mark for error checking method, 2 marks for description: Checksum • A value is calculated from the data // Description of calculation • Value is transmitted with data • Value is recalculated after transmission • If the values match the data is (more likely to be) accurate Parity check • A parity bit is transmitted with each byte of data • Odd or even (parity can be used) • Counts / checks number of 1’s // counts / checks to see if 1’s are even // counts / checks to see if 1’s are odd • (Each byte is) checked after transmission to see if it matches the odd/even parity used Automatic Repeat Request (ARQ) • Uses acknowledgement and timeout • When a device detects an error in data transmission it asks for the packet to be resent / no error detected, positive acknowledgment sent • The sending device resends the packet after the request to resend/ timeout received • This process is continuous until the packet received is correct/until the ARQ limit is reached Echo (check) • Copy of data is sent back to sender • Data is compared to see if it matches • If it does not match error detected	**6**

Question	**Answer**	**Marks**
5(a)	Any **four** from: • Data / files • Stored in a <u>text file</u> • Downloaded to a user's computer when a website is visited // webserver sends to web browser • Stored on a user's computer • Stored by a browser • Detected by the website when it is visited again	**4**
5(b)	Any **two** from: e.g. • To store personal information/data • To store login details • To save items in an online shopping basket • To track/save internet surfing habits // to track website traffic • To carry out targeted advertising • To store payment details • To customise a webpage // to store user preferences • Store progress in online games/quizzes	**2**

Question	**Answer**	**Marks**
6	1 mark for each correct term, in this order: • Interrupt • Compiler • ALU/Arithmetic and Logic Unit • ARQ/Automatic repeat request	**4**

Question	Answer	Marks
7	1 mark for each correct logic gate with the correct input(s) 	**7**

Question	Answer	Marks
8(a)	1 mark for correct calculation method, 1 mark for correct answer: • 2048/1024 (or 1024 × 2) • 2 GB	**2**
8(b)	• Instructions/programs/data • … currently in use	**2**
8(c)	Any **three** from: • RAM is volatile, ROM is non-volatile • RAM is temporary, ROM is (semi) permanent • RAM normally has a larger capacity than ROM • RAM can be edited ROM cannot be edited // Data can be read from and written to RAM, ROM can only be read from.	**3**

Question	Answer	Marks
9(a)	• It is an <u>input</u> device • It measures/takes (physical) readings of the surrounding environment / environment by example / physical properties	**2**
9(b)	1 mark for each sensor, 2 marks for each description: Moisture (sensor) • To measure the water content of the soil • To alert when the soil is too dry or too wet/needs watering pH (sensor) • To measure how acidic/alkaline the soil is • To alert when there may be something polluting the soil Light (sensor) • To measure the brightness of the environment • To alert when the fruit has too little/too much light Temperature (sensor) • To measure the temperature of the environment • To alert when it is too hot/too cold for the fruit to grow Gas (sensor) • To measure the amount of CO2/oxygen present • To alert when too much CO2/oxygen present Humidity (sensor) • To measure the water content in the air • To alert when the air is too dry Infra-red / motion (sensor) • To measure level of infra-red/microwaves deflected • To alert to any intruders e.g. animals stealing the fruit	**6**

Question	Answer	Marks
10(a)	Any **three** from: • It is a (security) protocol • It encrypts data (sent over the web/network) • It is the updated version of SSL • It has two layers • It has a handshake layer • It has a record layer	**3**
10(b)	1 mark for each correct application, examples could include: • Online banking • Online shopping // Online payment systems • Email • Cloud based storage • Intranet/extranet • VPN • VoIP • Instant messaging (IM) // social networking	**3**

Question	Answer	Marks
11	1 mark for each correct missing word, in the correct order: • Plagiarism • Free software • Freeware • Shareware • Ethics	**5**

Cambridge Assessment International Education
Cambridge International General Certificate of Secondary Education

---

COMPUTER SCIENCE **0478/13**

Paper 1 **October/November 2017**

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 2017 series for most Cambridge IGCSE®, Cambridge International A and AS Level components and some Cambridge O Level components.

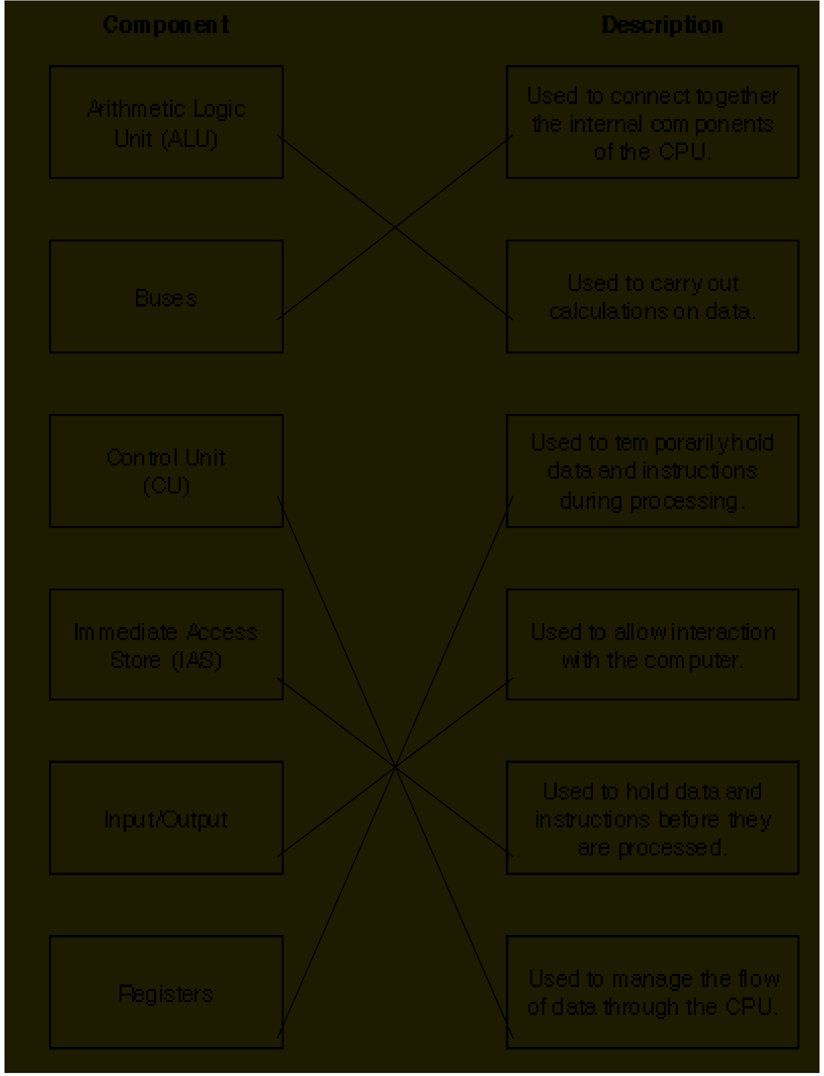
This syllabus is approved for use in England, Wales and Northern Ireland as a Cambridge International Level 1/Level 2 Certificate.

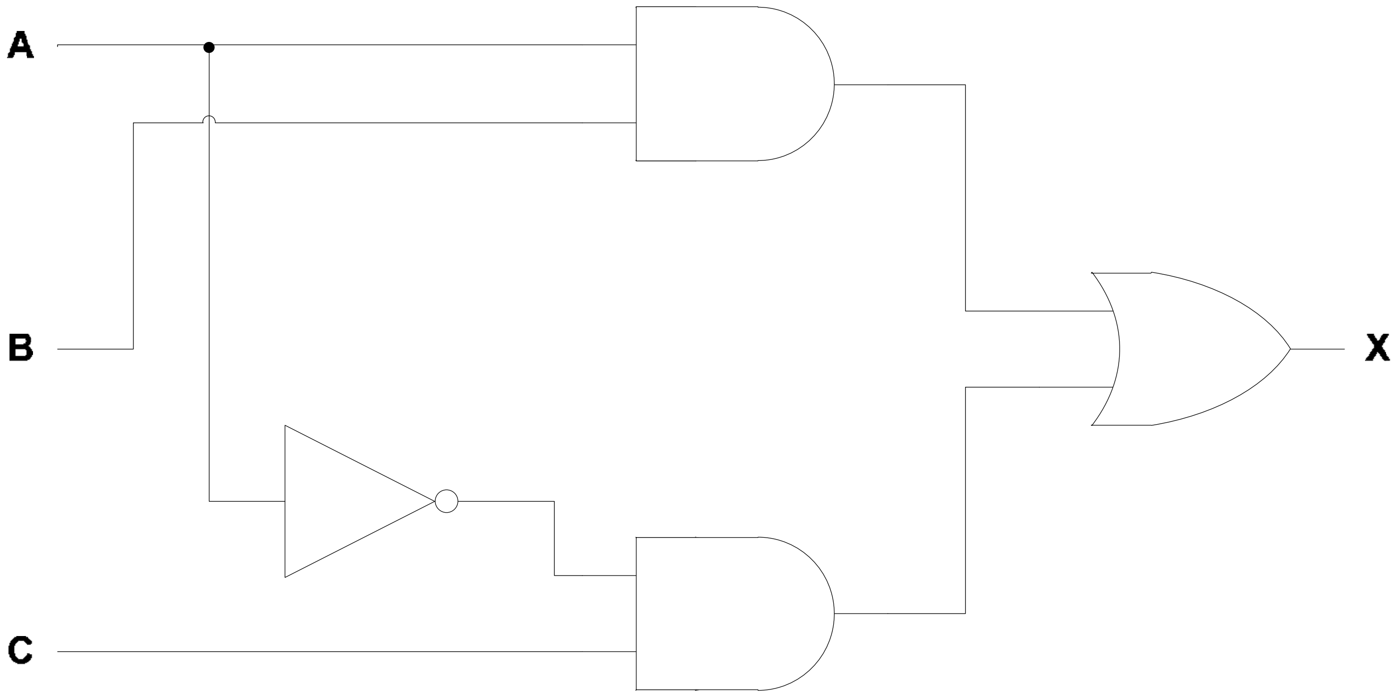
This document consists of **10** printed pages.

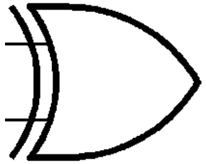
Question	Answer	Marks
1(a)	Output	**1**
1(b)	1 mark for each correct conversion *E* | *0* | *4* **1 1 1 0** | **0 0 0 0** | **0 1 0 0**	**3**
1(c)	Any **one** from: – Hexadecimal codes can fit in a smaller display rather than a full text based message – Smaller amount of memory needed to store the hex error messages than text based	**1**
1(d)	1 mark for correct sensor, 1 mark for corresponding use Possible examples could include: – Temperature (sensor) – To monitor the temperature of the water – Pressure (sensor) – To monitor the level of water in the washing machine – Motion (sensor) – To monitor whether the drum is still in motion – pH (sensor) – To monitor the level of water hardness/detergent present in the water	**6**

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>2</td><td>1 mark for each correct file format e.g.
<table>
<tr><th>File type</th><th>File format</th></tr>
<tr><td>Pictures</td><td>.JPEG</td></tr>
<tr><td>Text</td><td>.doc, .txt, .rtf,
.docx, .odt .pdf</td></tr>
<tr><td>Sound</td><td>.mp3, .wav, .aif,
.flac, .mid</td></tr>
<tr><td>Video</td><td>.mp4, .flv, .wmv</td></tr>
</table>
</td><td>3</td></tr>
</table>

Question	Answer	Marks
3(a)	– Part 1 (access) protocol – Part 2 domain (name) – Part 3 filename	**3**
3(b)	**Four** from: – IP address is used to identify a device (on the Internet / network) – IP address is allocated by the network/ ISP – Can be used in place of URL – IP addresses can be IPv4 or IPv6 – IP address can be static … – ... meaning it doesn't change each time it is connected to the Internet – IP address can be dynamic …. – … meaning that it can change each time a device is connected to the Internet – Any valid example (e.g. xxx.xxx.xxx.xxx or xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx)	**4**

Question	Answer	Marks
4	1 mark for each correct line up to a total of 5 marks 	**5**

Question	Answer	Marks
5(a)	1 mark for each correct logic gate  	**4**

Question	Answer	Marks
5(b)	1 mark for correct logic gate symbol:  Any **four** from: – similar to an OR gate – It has (at least) two inputs – Output will be high/1 if both inputs are different – Output will be high/1 if either input is high – Output will be low/0 if both inputs are high – Output will be low/0 if both inputs are low	**5**

Question	Answer	Marks
6	Any **six** from: 2D – (Scanner) shines a light onto the surface of a document // Light moves across document – Reflected light is captured – Uses mirrors and lenses – Captured image is converted into a digital file – Produces a 2D digital image 3D – Scanners shines a laser (or light) over the surface of a 3D object – Records measurements of the geometry/dimensions of the object – Measurements are converted to digital file – Produces a 3D digital model	**6**

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>7</td><td>1 mark for each correct tick
<table>
<tr><th>Statement</th><th>true (✓)</th><th>false (✓)</th></tr>
<tr><td>Firewalls can monitor incoming and outgoing traffic.</td><td>✓</td><td></td></tr>
<tr><td>Firewalls operate by checking traffic against a set of rules.</td><td>✓</td><td></td></tr>
<tr><td>Firewalls cannot block access to a certain website.</td><td></td><td>✓</td></tr>
<tr><td>Firewalls can be software and hardware.</td><td>✓</td><td></td></tr>
<tr><td>Firewalls can act as intermediary servers.</td><td></td><td>✓</td></tr>
<tr><td>Firewalls can block unauthorised traffic.</td><td>✓</td><td></td></tr>
</table>
</td><td>6</td></tr>
</table>

Question	Answer	Marks
8(a)	Any **three** from: – Human error (e.g. deleting/overwriting data) – Physical damage – Power failure/surge – Hardware failure – Software crashing	**3**
8(b)	Any **three** from: – Online shopping // Online payment systems // Online booking – Email – Cloud based storage – Intranet/extranet – VPN – VoIP // video conferencing – Instant messaging (IM) // social networking // online gaming	**3**

Question	Answer	Marks
8(c)	1 mark for identifying, 1 mark for description – Strong password – To make it difficult to hack an account – Biometric device – To use data that is difficult to fake as a password – TLS // Encryption – To make data meaningless if intercepted – To encrypt data that is exchanged (TLS only) – More secure than SSL (TLS only) – Anti-spyware (software) – To find and remove any spyware that is installed on a computer – To help stop key loggers recording key presses – Firewall – To help prevent unauthorised access to an account – Blocks any requests that do not meet/match the criteria – Authentication (card reader at home)/mobile security code app/two-step verification – To add another level of identification of the user – Use of drop-down boxes (or equivalent) – So key loggers cannot record the key presses – Proxy server – To divert an attack away from the main system	**6**

Question	Answer	Marks
9(a)	Any **four** from: – (Red) laser is used – (Laser beams) shines onto surface of the disk – It is rotated (at a constant speed) to be read – Surface is covered in a track (that spirals from the centre) – Data is represented on the surface using pits and lands – Pits and lands represent binary values – Pits reflect light back differently (to the area in between/land) – Optical device can determine the binary value from the light reflection	**4**
9(b)	1 mark for calculation, 1 mark for correct answer: – 1000 × 16 – 16000/8 – Answer is **2000** bytes	**2**
9(c)	**Four** from: (Max 2 for either primary or secondary) – Primary RAM and ROM – Secondary HDD and SSD – Primary is directly accessible by CPU – Secondary is not directly accessible by CPU – Primary is internal to computer – Secondary can be internal **or** external to the computer – Primary stores boot up instructions **and** can hold data whilst being processed – Secondary stores files/software – Primary has faster access speed – Secondary has a slower access speed – Primary has both volatile **and** non-volatile – Secondary is non-volatile	**4**

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>10</td><td>1 mark for each correct tick
<table>
<tr><th>Statement</th><th>true (✓)</th><th>false (✓)</th></tr>
<tr><td>Assembly language uses mnemonic codes.</td><td>✓</td><td></td></tr>
<tr><td>Assembly language programs do not need a translator to be executed.</td><td></td><td>✓</td></tr>
<tr><td>Assembly language is a low-level programming language.</td><td>✓</td><td></td></tr>
<tr><td>Assembly language is specific to the computer hardware.</td><td>✓</td><td></td></tr>
<tr><td>Assembly language is machine code.</td><td></td><td>✓</td></tr>
<tr><td>Assembly language is often used to create drivers for hardware.</td><td>✓</td><td></td></tr>
</table></td><td>6</td></tr>
</table>

Cambridge Assessment International Education
Cambridge International General Certificate of Secondary Education

COMPUTER SCIENCE **0478/21**

Paper 2 **October/November 2017**

MARK SCHEME

Maximum Mark: 50

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 2017 series for most Cambridge IGCSE®, Cambridge International A and AS Level components and some Cambridge O Level components.

This syllabus is approved for use in England, Wales and Northern Ireland as a Cambridge International Level 1/Level 2 Certificate.

This document consists of **7** printed pages.

Question	Answer	Marks

1(a)(i)	1 mark per bullet:
	• At least one array declaration
	• At least one array has an appropriate name
	• All arrays with appropriate names
	Many correct answers, they must be meaningful. These are examples only.
	`Array_2Seater[ ]`
	`Array_4Seater[ ]`
	`Array_Historic[ ]`
		3

1(a)(ii)	1 mark per bullet: • Name of variable • Purpose of variable • Name of constant • Purpose of constant Many correct answers, they must be meaningful. These are examples only. Variable `NumFlights` to store the number of flights in a day Constant `FlightCost2Seat30` to store the cost of a 30 minute flight in a 2 seater plane	**4**

Question	Answer	Marks
1(b)	Any **five** from: • Prompt for plane • Input plane • Prompt for another input length of flight along with the input. • Attempt at calculation of maximum number of flights in a day • Using correct values for maximum number of flights (from calculation or otherwise) • Calculation/determination of cost of a single flight for selected plane and duration • Calculation of income that can be generated for one combination of plane and flight • Output of total possible income for one combination of plane and flight with message(s) Algorithm example:	5

```
OUTPUT "Please Enter Type of Plane"
OUTPUT "1: 2 Seater"
OUTPUT "2: 4 Seater"
OUTPUT "3: Historic"
INPUT PlaneType
OUTPUT "Please Enter Length of Flight"
INPUT FlightLength
CASE FlightLength of
  30: OUTPUT "Maximum number of flights is 10"
  60: OUTPUT "Maximum number of flights is 6"
OTHERWISE OUTPUT "Invalid length of flight"
ENDCASE

CASE PlaneType of
  1: Price30 ← 100; Price60 ← 150
  2: Price30 ← 120; Price60 ← 200
  3: Price30 ← 120; Price60 ← 500
OTHERWISE OUTPUT "Invalid type of plane"
ENDCASE

CASE FlightLength of
  30: OUTPUT "Total Possible Income is ", Price30 * 10
  60: OUTPUT "Total Possible Income is ", Price60 * 6
ENDCASE
```

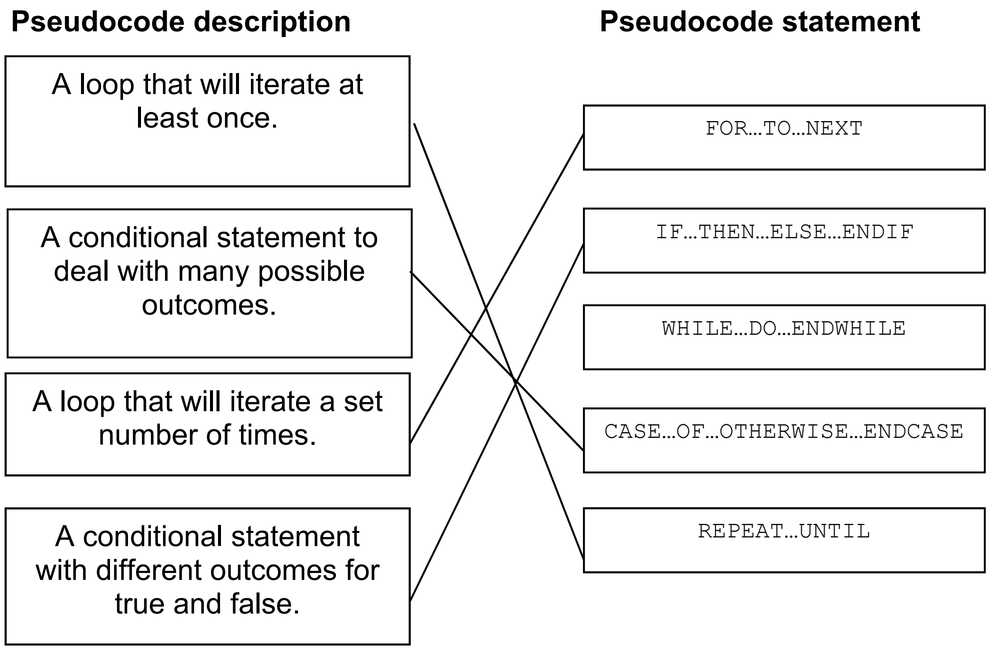
Question	Answer	Marks
1(c)	1 mark for each correct point related to the inputs for **Task 1** • Description of how the program would validate the input • Description/identification of input(s) • Type of validation check • Checking inputs against stored data/maxima/correct data • Dry-running the program • Use of test data • Identification of types of test data • Example(s) of test data	4

Question	Answer	Marks
1(d)	Any **four** from: • Input timeslot • Check 3 types of plane • Methodology for checking time slot • Identify any planes available • Output plane(s) available • Output if no planes available	**4**

Question	Answer	Marks
2	1 mark for each error identified plus suggested correction (the corrected lines must be written in full) Line 4 **correct line** `WHILE Number <= 99 OR Number > 1000` Line 7 **correct line** `Num[Index] = Number` Line 9 **correct line** `NEXT (Index)` Line 10 **correct line** `PRINT Count`	**4**

Question	Answer	Marks
3(a)	1 mark per bullet: • Validation checks whether data to be entered is possible/sensible // computer check • Verification checks that data entered is the data that was intended to be entered // can be a human check // matches the source	**2**
3(b)	1 mark for each valid point **Either** • Double Entry // suitable practical example • the data will be entered twice • compared by the computer or by a human • if a discrepancy is found, the data entry operator is asked to re-enter the data **Or** • Visual Verification // suitable practical example • the data will be compared to the source ‘document’ • compared by a human • if a discrepancy is found, the data is re-entered	**2**

Question	Answer	Marks
3(c)	1 mark for explanation and 1 mark for an expansion • Library routine is a list of instructions // block of code // subroutine • … that is used often … • … which is given a name • … and which can be called from other programs • Library routines make writing programs easier and faster as the code is already written • Library routines make program testing easier as the code has already been tested and debugged	**2**

Question	Answer	Marks
4(a)	1 mark for each correct line 	**4**
4(b)	1 mark per bullet: • Appropriate loop controls • Read from array • Print from array (the last two points can be in one statement) Note reading and printing **MUST** be within the same loop Example algorithm:	**3**

4(b) Example algorithm:

```
Count ← 0
WHILE Count < 50 DO
   OUTPUT Name[Count]
   Count ← Count + 1
ENDWHILE
```

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>5(a)</td><td>
<table>
<tr><th>Flag</th><th>Count</th><th>Name[1]</th><th>Name[2]</th><th>Name[3]</th><th>Name[4]</th><th>Temp</th></tr>
<tr><td></td><td></td><td>Jamal</td><td>Amir</td><td>Eve</td><td>Tara</td><td></td></tr>
<tr><td>0</td><td>1</td><td>Amir</td><td>Jamal</td><td>Eve</td><td>Tara</td><td>Jamal</td></tr>
<tr><td>1</td><td>2</td><td>Amir</td><td>Jamal</td><td>Eve</td><td>Tara</td><td>Jamal</td></tr>
<tr><td>1</td><td>3</td><td>Amir</td><td>Eve</td><td>Jamal</td><td>Tara</td><td>Jamal</td></tr>
<tr><td>1</td><td>4</td><td>Amir</td><td>Eve</td><td>Jamal</td><td>Tara</td><td>Jamal</td></tr>
<tr><td>0</td><td>1</td><td>Amir</td><td>Eve</td><td>Jamal</td><td>Tara</td><td>Jamal</td></tr>
<tr><td>0</td><td>2</td><td>Amir</td><td>Eve</td><td>Jamal</td><td>Tara</td><td>Jamal</td></tr>
<tr><td>0</td><td>3</td><td>Amir</td><td>Eve</td><td>Jamal</td><td>Tara</td><td>Jamal</td></tr>
<tr><td>0</td><td>4</td><td>Amir</td><td>Eve</td><td>Jamal</td><td>Tara</td><td>Jamal</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table>
(1 Mark) (1 Mark) (--------1 Mark--------) (--------1 Mark--------) (1 Mark)
</td><td>5</td></tr>
<tr><td>5(b)</td><td>1 mark per bullet:
Sorting the names
Ascending order / A to Z / lowest to highest / Alphabetic order</td><td>2</td></tr>
</table>

Question	Answer	Marks
6(a)	1 mark for any **sensible appropriate** field name 1 mark for data type, purpose + example data Example 1: Field Name: SPECIESID Data Type: Alphanumeric Purpose: Primary key Example Data: SP06583 Example 2: Field name: NUMBER Data Type: Integer Purpose: To record how many of that species there are at the park Example Data: 30	**2**

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>6(b)</td><td>
<table>
<tr><td>Field:</td><td>Species</td><td>Classification</td><td>Diet</td><td>Legs</td><td></td></tr>
<tr><td>Table:</td><td>LIVESTOCK</td><td>LIVESTOCK</td><td>LIVESTOCK</td><td>LIVESTOCK</td><td></td></tr>
<tr><td>Sort:</td><td>Ascending/
Descending</td><td></td><td></td><td></td><td></td></tr>
<tr><td>Show:</td><td>☑</td><td>☐</td><td>☐</td><td>☐</td><td>☐</td></tr>
<tr><td>Criteria:</td><td></td><td>“Mammal”</td><td>“Herbivore”</td><td>4</td><td></td></tr>
<tr><td>or:</td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td>(1 Mark)</td><td>(1 Mark)</td><td>(1 Mark)</td><td>(1 Mark)</td><td></td></tr>
</table>
1 mark per completely correct column.</td><td>4</td></tr>
</table>

Cambridge Assessment International Education
Cambridge International General Certificate of Secondary Education

---

COMPUTER SCIENCE	**0478/22**
Paper 2	**October/November 2017**
MARK SCHEME	
Maximum Mark: 50	

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 2017 series for most Cambridge IGCSE®, Cambridge International A and AS Level components and some Cambridge O Level components.

This syllabus is approved for use in England, Wales and Northern Ireland as a Cambridge International Level 1/Level 2 Certificate.

This document consists of **8** printed pages.

Question	Answer	Marks
1(a)(i)	1 mark for appropriate variable name, 1 mark for appropriate data type, 1 mark for appropriate use. Many correct answers, they must be meaningful. These are examples only. – `HireTotal,` integer, running total of money taken (for the day) – `HoursHired,` real, running total of hours hired for the day – `Returned,` real, hour and fraction of hour when next returned	**3**
1(a)(ii)	1 mark for appropriate constant name, 1 mark for appropriate value. Many correct answers, they must be meaningful. These are examples only. – `HourPrice, 20.00` – `HalfHourPrice 12.00`	**2**
1(b)	1 mark for validation check, all checks must be different, 1 mark for the reason and 1 mark for the test data. The only inputs for task 1 can be length of hire, money taken, time of hire and time of return. There are many possible correct answers these are examples only. Validation check – range check for time of hire Reason – cannot be hired before 10:00 returned after 17:00 Test data – 12:00, 19:00 Validation check – type check for money taken Reason – must be a numeric value Test data – 20.00, bob	**6**

Question	Answer	Marks
1(c)	– any loop for 10 boats (1 mark) **Four** from: – Initialisation – check `HoursHired` against `MaxHoursHired` … … store the `BoatNumber` … update `MaxHoursHired` if greater – check if `HoursHired = 0` … … if so add 1 to `NumberBoatsUnused` – update daily totals (for hours and money) – output report with messages (including totals for hours and money, and number of boats unused and the most used boat). Max 4 marks	**5**

Example:

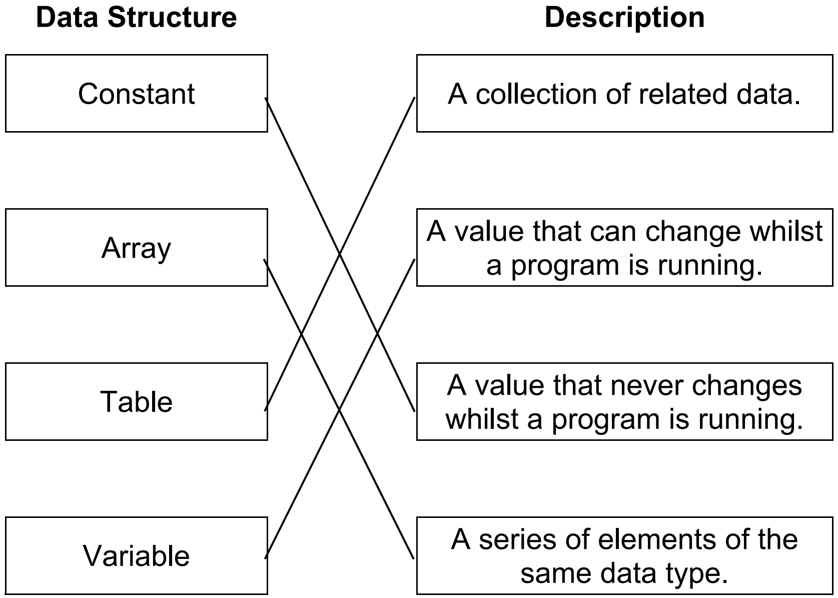
```
MaxHoursHired ← 0
TotalHoursHired ← 0
TotalMoney ← 0
NumberBoatsUnused ← 0
FOR BoatNumber ← 1 to 10
  TotalMoney ← TotalMoney + Money(BoatNumber)
  TotalHoursHired ← TotalHoursHired + HoursHired(BoatNumber)
  IF HoursHired(BoatNumber) = 0
    THEN NumberBoatsUnused ← NumberBoatsUnused + 1
  ENDIF
  IF HoursHired(BoatNumber) > MaxHoursHired
    THEN
     MostUsed ← BoatNumber
     MaxHoursHired ← HoursHired(BoatNumber)
  ENDIF
NEXT BoatNumber
PRINT "Boats were hired for ", TotalHoursHired, " hours"
PRINT "Total amount of money taken was ", TotalMoney
PRINT NumberBoatsUnused, " boats were not used"
Print "Boat number ", MostUsed, " was used most"
```

Question	Answer	Marks
1(d)	Maximum 4 marks in total for question part e.g. **Explanation** (may include reference to program statements) – check all boats for… – … return time < current time // current booking slot available or return time > current time// current booking slot not available – keep a running total of those available – display number of boats Example:	**4**

```
FOR BoatNumber ← 1 to 10    loop to check for all boats
  IF ReturnTime(BoatNumber) <= CurrentTime  check return time against current time
    THEN BoatsAvailable ← BoatsAvailable + 1     keep a running total
  ENDIF
NEXT BoatNumber
PRINT "Number of boats available ", BoatsAvailable     display number of boats
```

Question	Answer	Marks
2	1 mark for each, there may be other solutions, award full marks for any working solution any **six** from: initialise total (outside loop) Input number of numbers (outside loop with validation) Loop using input value Input number (inside loop) Update Total (inside loop) Calculate average Print average and total (outside loop) Sample algorithm:	**6**

```
INPUT NumberCount
Total ← 0
FOR Count ← 1 TO NumberCount
  INPUT Number
  Total ← Total + Number
NEXT
Average ← Total/NumberCount
  PRINT Total, Average
```

Question	Answer	Marks
3	1 mark for each correct line, max **3** marks. 	**3**

Question	Answer	Marks

4	2 marks for identification, 1 mark for description, 1 mark for reason.
	
	Identification:
	`CASE …`
	`… OF … OTHERWISE … (ENDCASE)` or
	`… OF … (OTHERWISE) … ENDCASE`
	
	Description:
	– a statement that allows for multiple selections // not any of the above
	
	Reason:
	– to simplify pseudocode/ make pseudocode more understandable etc.
		4

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>5(a)</td><td>
<table>
<tr><th>Accept</th><th>Reject</th><th>Count</th><th>Sack</th><th>OUTPUT</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td></td><td></td></tr>
<tr><td>1</td><td></td><td>1</td><td>50.4</td><td></td></tr>
<tr><td>2</td><td></td><td>2</td><td>50.3</td><td></td></tr>
<tr><td></td><td>1</td><td>3</td><td>49.1</td><td></td></tr>
<tr><td>3</td><td></td><td>4</td><td>50.3</td><td></td></tr>
<tr><td>4</td><td></td><td>5</td><td>50.0</td><td></td></tr>
<tr><td>5</td><td></td><td>6</td><td>49.5</td><td></td></tr>
<tr><td>6</td><td></td><td>7</td><td>50.2</td><td></td></tr>
<tr><td>7</td><td></td><td>8</td><td>50.3</td><td></td></tr>
<tr><td>8</td><td></td><td>9</td><td>50.5</td><td></td></tr>
<tr><td></td><td>2</td><td>10</td><td>50.6</td><td>8 2</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>← (1 mark) →</td><td>← (1 mark) →</td><td>← (1 mark) →</td><td>← (1 mark) →</td><td>← (1 mark) →</td></tr>
</table>
</td><td>5</td></tr>
<tr><td>5(b)</td><td>– change to <code>Is Count = 50?</code>
– remove <code>IS Sack > 50.5?</code></td><td>2</td></tr>
</table>

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>6(a)</td><td>– 1 mark for each field suitable name, 1 mark for appropriate data type and appropriate data sample
The following are examples there are many different correct answers.
– Engine Number, text, 21012
– Class, text, P6
– Service Date, date, 4/3/2017</td><td>6</td></tr>
<tr><td>6(b)</td><td>– Engine Number // Correct field number</td><td>1</td></tr>
<tr><td>6(c)</td><td>
<table>
<tr><td>Field:</td><td>Engine Number</td><td>Class</td><td>Service Date</td><td></td></tr>
<tr><td>Table:</td><td>TRAIN</td><td>TRAIN</td><td>TRAIN</td><td></td></tr>
<tr><td>Sort:</td><td></td><td></td><td></td><td></td></tr>
<tr><td>Show:</td><td>☑</td><td>☐</td><td>☐</td><td>☐</td></tr>
<tr><td>Criteria:</td><td></td><td>Like ‘P*’ // Like ‘P?’</td><td><10/11/2016</td><td></td></tr>
<tr><td>or:</td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td>(1 mark)</td><td>(1 mark)</td><td>(1 mark)</td><td></td></tr>
</table>
</td><td>3</td></tr>
</table>

Cambridge Assessment International Education
Cambridge International General Certificate of Secondary Education

COMPUTER SCIENCE **0478/23**

Paper 2 **October/November 2017**

MARK SCHEME

Maximum Mark: 50

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 2017 series for most Cambridge IGCSE®, Cambridge International A and AS Level components and some Cambridge O Level components.

This syllabus is approved for use in England, Wales and Northern Ireland as a Cambridge International Level 1/Level 2 Certificate.

This document consists of **7** printed pages.

Question	Answer	Marks

1(a)(i)	1 mark per bullet:
	• At least one array declaration
	• At least one array has an appropriate name
	• All arrays with appropriate names
	Many correct answers, they must be meaningful. These are examples only.
	`Array_2Seater[ ]`
	`Array_4Seater[ ]`
	`Array_Historic[ ]`
		3

1(a)(ii)	1 mark per bullet: • Name of variable • Purpose of variable • Name of constant • Purpose of constant Many correct answers, they must be meaningful. These are examples only. Variable `NumFlights` to store the number of flights in a day Constant `FlightCost2Seat30` to store the cost of a 30 minute flight in a 2 seater plane	**4**

Question	Answer	Marks
1(b)	Any **five** from: • Prompt for plane • Input plane • Prompt for another input length of flight along with the input. • Attempt at calculation of maximum number of flights in a day • Using correct values for maximum number of flights (from calculation or otherwise) • Calculation/determination of cost of a single flight for selected plane and duration • Calculation of income that can be generated for one combination of plane and flight • Output of total possible income for one combination of plane and flight with message(s) Algorithm example:	5

```
OUTPUT "Please Enter Type of Plane"
OUTPUT "1: 2 Seater"
OUTPUT "2: 4 Seater"
OUTPUT "3: Historic"
INPUT PlaneType
OUTPUT "Please Enter Length of Flight"
INPUT FlightLength
CASE FlightLength of
  30: OUTPUT "Maximum number of flights is 10"
  60: OUTPUT "Maximum number of flights is 6"
OTHERWISE OUTPUT "Invalid length of flight"
ENDCASE

CASE PlaneType of
  1: Price30 ← 100; Price60 ← 150
  2: Price30 ← 120; Price60 ← 200
  3: Price30 ← 120; Price60 ← 500
OTHERWISE OUTPUT "Invalid type of plane"
ENDCASE

CASE FlightLength of
  30: OUTPUT "Total Possible Income is ", Price30 * 10
  60: OUTPUT "Total Possible Income is ", Price60 * 6
ENDCASE
```

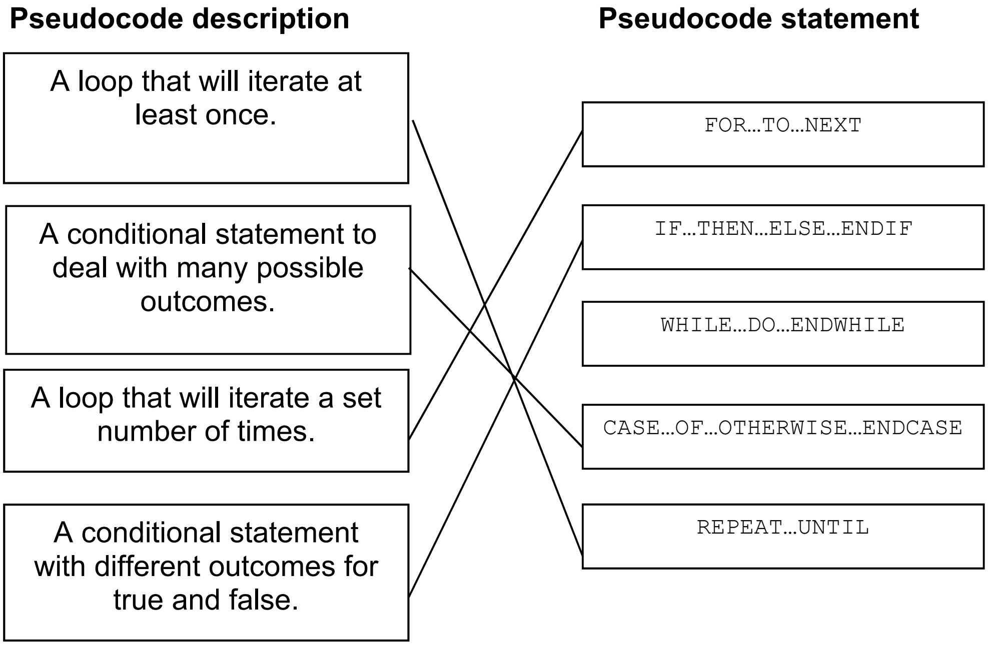
Question	Answer	Marks
1(c)	1 mark for each correct point related to the inputs for **Task 1** • Description of how the program would validate the input • Description/identification of input(s) • Type of validation check • Checking inputs against stored data/maxima/correct data • Dry-running the program • Use of test data • Identification of types of test data • Example(s) of test data	4

Question	Answer	Marks
1(d)	Any **four** from: • Input timeslot • Check 3 types of plane • Methodology for checking time slot • Identify any planes available • Output plane(s) available • Output if no planes available	**4**

Question	Answer	Marks
2	1 mark for each error identified plus suggested correction (the corrected lines must be written in full) Line 4 **correct line** `WHILE Number <= 99 OR Number > 1000` Line 7 **correct line** `Num[Index] = Number` Line 9 **correct line** `NEXT (Index)` Line 10 **correct line** `PRINT Count`	**4**

Question	Answer	Marks
3(a)	1 mark per bullet: • Validation checks whether data to be entered is possible/sensible // computer check • Verification checks that data entered is the data that was intended to be entered // can be a human check // matches the source	**2**
3(b)	1 mark for each valid point **Either** • Double Entry // suitable practical example • the data will be entered twice • compared by the computer or by a human • if a discrepancy is found, the data entry operator is asked to re-enter the data **Or** • Visual Verification // suitable practical example • the data will be compared to the source 'document' • compared by a human • if a discrepancy is found, the data is re-entered	**2**

Question	Answer	Marks
3(c)	1 mark for explanation and 1 mark for an expansion • Library routine is a list of instructions // block of code // subroutine • … that is used often … • … which is given a name • … and which can be called from other programs • Library routines make writing programs easier and faster as the code is already written • Library routines make program testing easier as the code has already been tested and debugged	2

Question	Answer	Marks
4(a)	1 mark for each correct line  	4
4(b)	1 mark per bullet: • Appropriate loop controls • Read from array • Print from array (the last two points can be in one statement) Note reading and printing **MUST** be within the same loop Example algorithm:	3

```
Count ← 0
WHILE Count < 50 DO
   OUTPUT Name[Count]
   Count ← Count + 1
ENDWHILE
```

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>5(a)</td><td>
<table>
<tr><th>Flag</th><th>Count</th><th>Name[1]</th><th>Name[2]</th><th>Name[3]</th><th>Name[4]</th><th>Temp</th></tr>
<tr><td></td><td></td><td>Jamal</td><td>Amir</td><td>Eve</td><td>Tara</td><td></td></tr>
<tr><td>0</td><td>1</td><td>Amir</td><td>Jamal</td><td>Eve</td><td>Tara</td><td>Jamal</td></tr>
<tr><td>1</td><td>2</td><td>Amir</td><td>Jamal</td><td>Eve</td><td>Tara</td><td>Jamal</td></tr>
<tr><td>1</td><td>3</td><td>Amir</td><td>Eve</td><td>Jamal</td><td>Tara</td><td>Jamal</td></tr>
<tr><td>1</td><td>4</td><td>Amir</td><td>Eve</td><td>Jamal</td><td>Tara</td><td>Jamal</td></tr>
<tr><td>0</td><td>1</td><td>Amir</td><td>Eve</td><td>Jamal</td><td>Tara</td><td>Jamal</td></tr>
<tr><td>0</td><td>2</td><td>Amir</td><td>Eve</td><td>Jamal</td><td>Tara</td><td>Jamal</td></tr>
<tr><td>0</td><td>3</td><td>Amir</td><td>Eve</td><td>Jamal</td><td>Tara</td><td>Jamal</td></tr>
<tr><td>0</td><td>4</td><td>Amir</td><td>Eve</td><td>Jamal</td><td>Tara</td><td>Jamal</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table>
(1 Mark) (1 Mark) (--------1 Mark--------) (--------1 Mark--------) (1 Mark)
</td><td>5</td></tr>
<tr><td>5(b)</td><td>1 mark per bullet:
• Sorting the names
• Ascending order / A to Z / lowest to highest / Alphabetic order</td><td>2</td></tr>
</table>

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>6(a)</td><td>1 mark for any **sensible appropriate** field name
1 mark for data type, purpose + example data

Example 1:
Field Name: SPECIESID
Data Type: Alphanumeric
Purpose: Primary key
Example Data: SP06583

Example 2:
Field name: NUMBER
Data Type: Integer
Purpose: To record how many of that species there are at the park
Example Data: 30</td><td>2</td></tr>
</table>

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>6(b)</td><td>
<table>
<tr><td>Field:</td><td>Species</td><td>Classification</td><td>Diet</td><td>Legs</td><td></td></tr>
<tr><td>Table:</td><td>LIVESTOCK</td><td>LIVESTOCK</td><td>LIVESTOCK</td><td>LIVESTOCK</td><td></td></tr>
<tr><td>Sort:</td><td>Ascending/ Descending</td><td></td><td></td><td></td><td></td></tr>
<tr><td>Show:</td><td>☑</td><td>☐</td><td>☐</td><td>☐</td><td>☐</td></tr>
<tr><td>Criteria:</td><td></td><td>“Mammal”</td><td>“Herbivore”</td><td>4</td><td></td></tr>
<tr><td>or:</td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td>(1 Mark)</td><td>(1 Mark)</td><td>(1 Mark)</td><td>(1 Mark)</td><td></td></tr>
</table>
1 mark per completely correct column.</td><td>4</td></tr>
</table>

Cambridge IGCSE

Cambridge International Examinations
Cambridge International General Certificate of Secondary Education

---

COMPUTER SCIENCE **0478/21**

Paper 2 Problem-solving and Programming **October/November 2017**

PRE-RELEASE MATERIAL

No Additional Materials are required.

This material should be given to the relevant teachers and candidates as soon as it has been received at the Centre.

READ THESE INSTRUCTIONS FIRST

Candidates should use this material in preparation for the examination. Candidates should attempt the practical programming tasks using their chosen high-level, procedural programming language.

The syllabus is approved for use in England, Wales and Northern Ireland as a Cambridge International Level 1/Level 2 Certificate.

This document consists of **2** printed pages.

DC (RW) 131976/2

CAMBRIDGE
International Examinations

[Turn over

In preparation for the examination candidates should attempt the following practical tasks by **writing and testing a program or programs**.

A small airfield operates a flying club where people can take a short flight to see if they would like flying lessons. The owner of the airfield has asked you to write a program to organise the flight bookings. The airfield operates three different planes and offers either a 30 minute or a 60 minute flight.

The following table shows the tariff:

Length of Flight	**2 Seater Plane**	**4 Seater Plane**	**Historic Plane**
30 minutes	$100	$120	$300
60 minutes	$150	$200	$500

After each flight, 30 minutes must be allowed for refuelling and safety checks before the next flight can take off. All planes offer both 30 minute and 60 minute flights, but, for the purpose of this activity, they will not be mixed on a given day, e.g. the 2 seater will offer 30 minute flights ONLY on one day and 60 minute flights ONLY on another day.

Write and test a program for the owner of the airfield.

- Your program must include appropriate prompts for the entry of data.
- Error messages and other output need to be set out clearly and understandably.
- All variables, constants and other identifiers must have meaningful names.

You will need to complete these **three** tasks. Each task must be fully tested.

TASK 1 – Work out the maximum income.

Assume that the flights take place between 08:00 and 18:00. Write a program that will work out the maximum income that can be generated by each plane in a day for each type of flight. The program should allow you to:

- input the type of plane
- input the length of flight
- calculate the maximum number of flights in a day
- output the total possible income per day for the choice of plane and length of flight

TASK 2 – Record bookings.

Write a program to store bookings for each plane and to allow you to find which planes are available at a given time slot during the day. The program should calculate the actual number of flights taken by each plane in that day.

TASK 3 – Work out income.

Modify TASK 2 so that it will calculate the total amount of money taken in a day for each plane, as well as the overall daily total for all three planes, and output the results.

Cambridge International Examinations
Cambridge International General Certificate of Secondary Education

---

COMPUTER SCIENCE **0478/22**

Paper 2 Problem-solving and Programming **October/November 2017**

PRE-RELEASE MATERIAL

No Additional Materials are required.

This material should be given to the relevant teachers and candidates as soon as it has been received at the Centre.

READ THESE INSTRUCTIONS FIRST

Candidates should use this material in preparation for the examination. Candidates should attempt the practical programming tasks using their chosen high-level, procedural programming language.

The syllabus is approved for use in England, Wales and Northern Ireland as a Cambridge International Level 1/Level 2 Certificate.

This document consists of **2** printed pages.

CAMBRIDGE
International Examinations

In preparation for the examination candidates should attempt the following practical tasks by **writing and testing a program or programs**.

The owner of a river boat hire company wants to calculate the daily profits from hiring out 10 rowing boats on the river. Boats are numbered 1 to 10. Boats can be hired for use between 10:00 and 17:00 every day.

Write and test a program for the owner.

- Your program must include appropriate prompts for the entry of data.
- Error messages and other output need to be set out clearly and understandably.
- All variables, constants and other identifiers must have meaningful names.

You will need to complete these **three** tasks. Each task must be fully tested.

TASK 1 – calculate the money taken in a day for one boat.

The cost of hiring a boat is $20 for one hour or $12 for half an hour. When a boat is hired the payment is added to the money taken for the day. The running total of hours hired that day is updated and the time when the boat must be returned is stored. At the end of the day the money taken and the total hours hired is output.
No boat can be hired before 10:00 or returned after 17:00.

TASK 2 – find the next boat available.

Extend TASK 1 to work for all 10 rowing boats. Use the data stored for each boat to find out how many boats are available for hire at the current time. If no boats are available show the earliest time that a boat will be available for hire.

TASK 3 – calculate the money taken for all the boats at the end of the day.

At the end of the day use the data stored for each boat to calculate the total amount of money taken and the total number of hours boats were hired that day. Find out how many boats were not used that day and which boat was used the most. Provide a report for the owner to show this information.

Cambridge International Examinations
Cambridge International General Certificate of Secondary Education

COMPUTER SCIENCE **0478/23**

Paper 2 Problem-solving and Programming **October/November 2017**

PRE-RELEASE MATERIAL

No Additional Materials are required.

This material should be given to the relevant teachers and candidates as soon as it has been received at the Centre.

READ THESE INSTRUCTIONS FIRST

Candidates should use this material in preparation for the examination. Candidates should attempt the practical programming tasks using their chosen high-level, procedural programming language.

The syllabus is approved for use in England, Wales and Northern Ireland as a Cambridge International Level 1/Level 2 Certificate.

This document consists of **2** printed pages.

DC (LK) 150376

In preparation for the examination candidates should attempt the following practical tasks by **writing and testing a program or programs**.

A small airfield operates a flying club where people can take a short flight to see if they would like flying lessons. The owner of the airfield has asked you to write a program to organise the flight bookings. The airfield operates three different planes and offers either a 30 minute or a 60 minute flight.

The following table shows the tariff:

Length of Flight	**2 Seater Plane**	**4 Seater Plane**	**Historic Plane**
30 minutes	$100	$120	$300
60 minutes	$150	$200	$500

After each flight, 30 minutes must be allowed for refuelling and safety checks before the next flight can take off. All planes offer both 30 minute and 60 minute flights, but, for the purpose of this activity, they will not be mixed on a given day, e.g. the 2 seater will offer 30 minute flights ONLY on one day and 60 minute flights ONLY on another day.

Write and test a program for the owner of the airfield.

- Your program must include appropriate prompts for the entry of data.
- Error messages and other output need to be set out clearly and understandably.
- All variables, constants and other identifiers must have meaningful names.

You will need to complete these **three** tasks. Each task must be fully tested.

TASK 1 – Work out the maximum income.

Assume that the flights take place between 08:00 and 18:00. Write a program that will work out the maximum income that can be generated by each plane in a day for each type of flight. The program should allow you to:

- input the type of plane
- input the length of flight
- calculate the maximum number of flights in a day
- output the total possible income per day for the choice of plane and length of flight

TASK 2 – Record bookings.

Write a program to store bookings for each plane and to allow you to find which planes are available at a given time slot during the day. The program should calculate the actual number of flights taken by each plane in that day.

TASK 3 – Work out income.

Modify TASK 2 so that it will calculate the total amount of money taken in a day for each plane, as well as the overall daily total for all three planes, and output the results.

Cambridge International Examinations
Cambridge International General Certificate of Secondary Education

CANDIDATE NAME	

CENTRE NUMBER						CANDIDATE NUMBER					

COMPUTER SCIENCE **0478/11**

Paper 1 Theory **October/November 2017**

1 hour 45 minutes

Candidates answer on the Question Paper.

No Additional Materials are required.

No calculators allowed.

READ THESE INSTRUCTIONS FIRST

Write your Centre number, candidate number and name in the spaces at the top of this page.
Write in dark blue or black pen.
You may use an HB pencil for any diagrams, graphs or rough working.
Do not use staples, paper clips, glue or correction fluid.
DO **NOT** WRITE IN ANY BARCODES.

Answer **all** questions.
No marks will be awarded for using brand names of software packages or hardware.

At the end of the examination, fasten all your work securely together.
The number of marks is given in brackets [ ] at the end of each question or part question.

The maximum number of marks is 75.

The syllabus is approved for use in England, Wales and Northern Ireland as a Cambridge International Level 1/Level 2 Certificate.

This document consists of **11** printed pages and **1** blank page.

DC (NH/SW) 134256/4

1 A washing machine has a small display screen built into it.

One use of the display screen is to show an error code when a problem has occurred with a washing cycle.

(a) State whether the display screen is an **input**, **output** or **storage device**.

....[1]

(b) The display screen shows a hexadecimal error code:

E04

This error code means that the water will not empty out of the washing machine.

Convert this error code to binary.

[3]

(c) State why hexadecimal is used to display the error code.

....

....

....

....[1]

(d) Identify **three** sensors that could be used in the washing machine.

State what each sensor could be used for.

Sensor 1 ....

Use ....

....

Sensor 2 ....

Use ....

....

Sensor 3 ....

Use ....

....

[6]

2 Data files are stored in different file formats.

Complete the table by providing a suitable file format for each file type. The first one has been done for you.

File type	File format
Pictures	.JPEG
Text	
Sound	
Video	

[3]

3 **(a)** An example of a Uniform Resource Locator (URL) is:

$$\underbrace{\textbf{http://}}_{\text{Part 1}}\textbf{www.}\underbrace{\textbf{cie.org.uk}}_{\text{Part 2}}\underbrace{\textbf{/index.htm}}_{\text{Part 3}}$$

Identify the **three** parts that make up this URL.

Part 1 ....

Part 2 ....

Part 3 ....

[3]

(b) Describe what is meant by an Internet Protocol (IP) address.

....

....

....

....

....

....

....

....[4]

4 **Six** components of a computer system and **six** descriptions are shown.

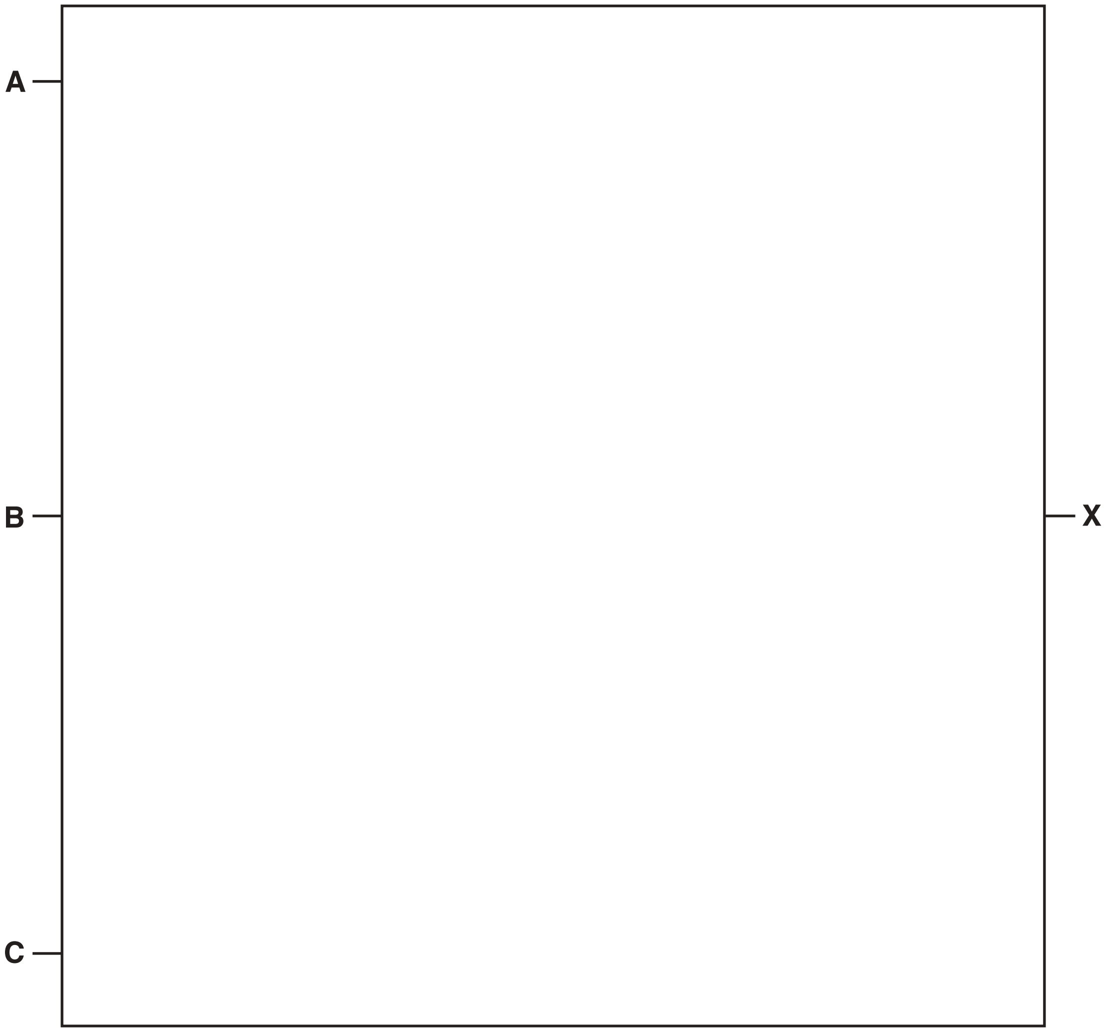
Draw a line to match each component with the most suitable description.

Component	**Description**
Arithmetic Logic Unit (ALU)	Used to connect together the internal components of the CPU.
Buses	Used to carry out calculations on data.
Control Unit (CU)	Used to temporarily hold data and instructions during processing.
Immediate Access Store (IAS)	Used to allow interaction with the computer.
Input/Output	Used to hold data and instructions before they are processed.
Registers	Used to manage the flow of data through the CPU.

[5]

5 **(a)** Draw a logic circuit for the logic statement:

X = 1 if ((A is 1 AND B is 1) OR (A is NOT 1 AND C is 1))



[4]

(b) Draw the symbol for an **XOR** gate and explain the function of this logic gate.

Explanation ....

....

....

....

....

....[5]

6 Describe the operation of a 2D scanner and a 3D scanner.

2D ....

....

....

....

....

....

3D ....

....

....

....

....

....

[6]

7 **Six** statements about firewalls are shown.

Tick (✓) to show whether each statement is **true** or **false**.

Statement	**true** (✓)	**false** (✓)
Firewalls can monitor incoming and outgoing traffic.		
Firewalls operate by checking traffic against a set of rules.		
Firewalls cannot block access to a certain website.		
Firewalls can be software and hardware.		
Firewalls can act as intermediary servers.		
Firewalls can block unauthorised traffic.		

[6]

8 **(a)** Data is valuable. It needs to be kept secure and it can easily be damaged.

Give **three** different ways that data can be accidentally damaged.

1 ....

....

2 ....

....

3 ....

....
[3]

(b) The Secure Socket Layer (SSL) protocol can be used to securely transmit data in online banking.

State **three** other different applications that use SSL.

Application 1 ....

Application 2 ....

Application 3 ....
[3]

(c) Online banking is increasing in popularity.

Online banking can be a risk as it can raise a number of security issues. SSL can be used as a security method to make online banking safer.

Identify and describe **three** other security methods that could be used to make online banking safer.

Security method 1 ....

....

....

....

Security method 2 ....

....

....

....

Security method 3 ....

....

....

....

[6]

9 **(a)** Optical storage media can be used to store data.

Describe how the data is read from a Compact Disc (CD).

....

....

....

....

....

....

....

....[4]

(b) Kamil wants to store a 16-bit colour image file. The image size is 1000 pixels.

Calculate the size of the file.

Give your answer in **bytes**. Show your working.

Working ....

....

....

....

Answer .... **bytes**

[2]

(c) Describe the differences between primary and secondary storage.

....

....

....

....

....

....

....

....[4]

10 **Six** statements about assembly language are shown.

Tick (✓) whether the statement is **true** or **false**.

Statement	**true** (✓)	**false** (✓)
Assembly language uses mnemonic codes.		
Assembly language programs do not need a translator to be executed.		
Assembly language is a low-level programming language.		
Assembly language is specific to the computer hardware.		
Assembly language is machine code.		
Assembly language is often used to create drivers for hardware.		

[6]

BLANK PAGE

Cambridge International Examinations
Cambridge International General Certificate of Secondary Education

CANDIDATE NAME			
CENTRE NUMBER		CANDIDATE NUMBER	

0049796571

COMPUTER SCIENCE **0478/12**

Paper 1 Theory **October/November 2017**

1 hour 45 minutes

Candidates answer on the Question Paper.

No Additional Materials are required.

No calculators allowed.

READ THESE INSTRUCTIONS FIRST

Write your Centre number, candidate number and name on all the work you hand in.
Write in dark blue or black pen.
You may use an HB pencil for any diagrams, graphs or rough working.
Do not use staples, paper clips, glue or correction fluid.
DO **NOT** WRITE IN ANY BARCODES.

Answer **all** questions.
No marks will be awarded for using brand names of software packages or hardware.

At the end of the examination, fasten all your work securely together.
The number of marks is given in brackets [ ] at the end of each question or part question.

The maximum number of marks is 75.

The syllabus is approved for use in England, Wales and Northern Ireland as a Cambridge International Level 1/Level 2 Certificate.

This document consists of **12** printed pages.

DC (NH/SW) 138025/2

CAMBRIDGE
International Examinations

1 A robot arm in a factory is programmed to move products.

The binary instructions to operate the robot arm are:

Operation	**Binary Instruction**			
UP	1	1	1	1
DOWN	0	0	0	1
LEFT	1	0	0	1
RIGHT	0	1	1	0
OPEN	1	1	0	0
CLOSE	0	0	1	1

The instructions are entered as hexadecimal values.

An operator enters the values:

9 1 C 3 F

Convert the values and write down the operation (e.g. RIGHT) carried out by the robot arm.

9 ....

1 ....

C ....

3 ....

F ....

[5]

2 Storage devices and storage media can be categorised as primary, secondary or off-line.

Write **primary**, **secondary** or **off-line** next to each storage device or medium to indicate its most suitable category.

HDD ....

RAM ....

ROM ....

CD-ROM ....

SSD ....

DVD-RAM ....

[6]

3 **(a)** Explain the differences between the binary number system and the denary number system.

....

....

....

....

....

....

....

....[4]

(b) Explain the process of converting the binary number 1010 into a denary number.

....

....

....

....

....

....

....

....

....

....[5]

[Turn over

4 A file server is used as a central data store for a network of computers.

Rory sends data from his computer to a file server that is approximately 100 metres away.

It is important that the data is transmitted accurately. Rory needs to be able to read data from and write data to the file server at the same time.

(a) **(i)** Use ticks (✓) to identify the most suitable data transmission methods for this application.

Method 1	Tick (✓)	**Method 2**	Tick (✓)
Serial		Simplex	
Parallel		Half-duplex	
		Duplex	

[2]

(ii) Explain why your answer to **part (a)(i)** is the most suitable data transmission.

....

....

....

....

....

....

....

....[4]

(b) Identify and describe **two** methods of error checking that can be used to make sure that the data stored after transmission is accurate.

Method 1 ....

....

....

....

....

....

....

....

Method 2 ....

....

....

....

....

....

....

....

[6]

5 Raj is using the Internet to do some online shopping. He visits a website that tells him that it uses cookies.

(a) Explain what is meant by the term **cookies**.

....

....

....

....

....

....

....

....[4]

(b) Give **two** examples of the use of cookies.

Example 1 ....

....

....

Example 2 ....

....

....

[2]

6 Selma writes the following **four** answers in her Computer Science examination.

State which computer terms she is describing.

"It is a signal. When the signal is received it tells the operating system that an event has occurred."

Selma is describing ....

"It takes source code written in a high level language and translates it into machine code. It translates the whole of the source code at once."

Selma is describing ....

"The part of the central processing unit (CPU) that carries out calculations."

Selma is describing ....

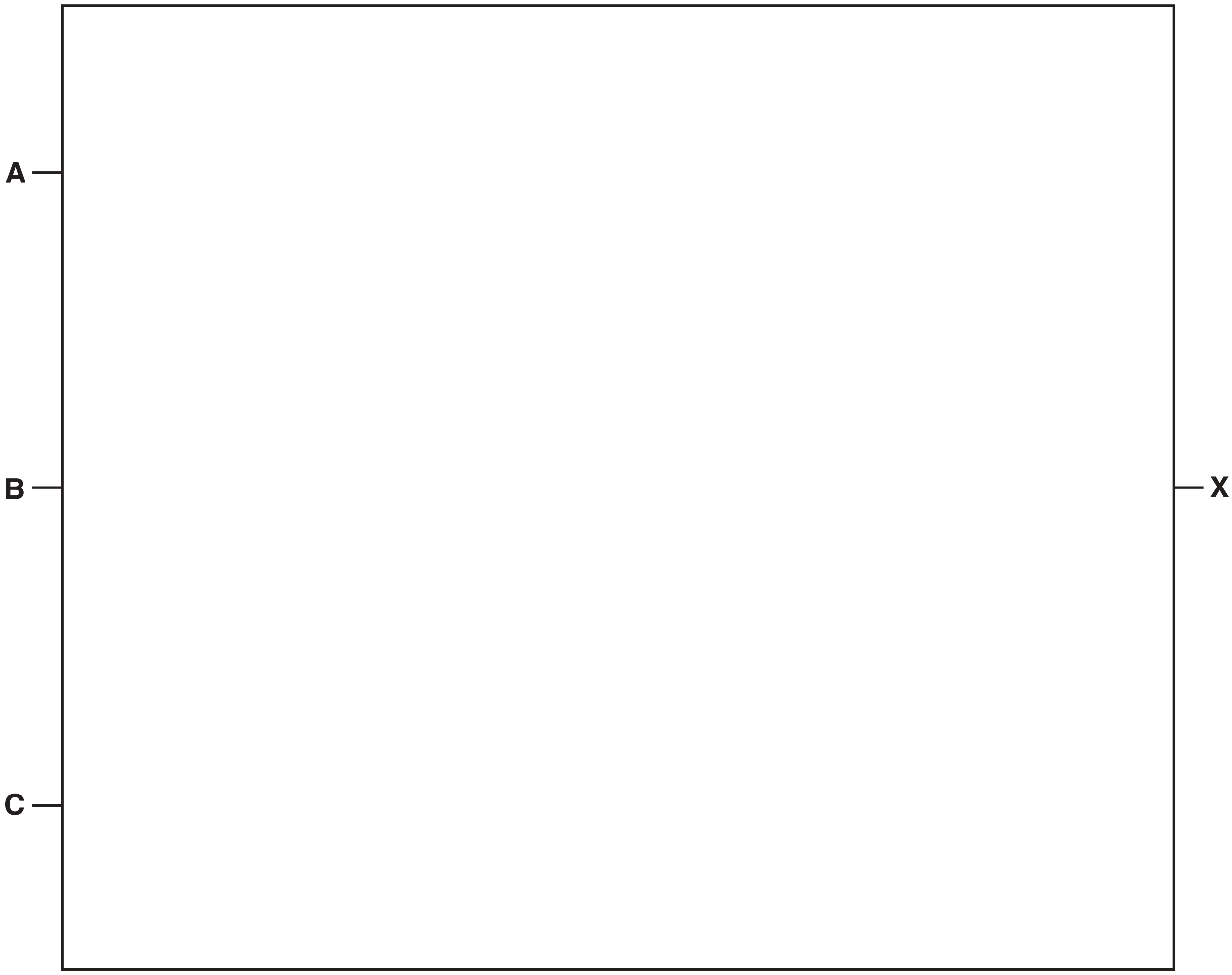
"When data is transmitted, if an error is detected in the data received a signal is sent to ask for the data to be retransmitted. This continues until the data received is correct."

Selma is describing ....

[4]

7 Draw a logic circuit to represent the logic statement:

X = 1 if (A is NOT 1 AND B is 1) AND (A is NOT 1 AND C is NOT 1) OR (B is 1 AND C is 1)



[7]

8 **(a)** A computer has 2048 MB of RAM.

How many GB of RAM does the computer have?

Show your working.

....

....

....

..............................GB [2]

(b) Describe **one** item that is stored in RAM.

....

....

....[2]

(c) Explain **three** ways that RAM is different to ROM.

1 ....

....

....

2 ....

....

....

3 ....

....

....

[3]

[Turn over

9 Anna has a farm that grows fruit.

She has a system that monitors the conditions for growing the fruit.

Sensors are used in this system.

(a) Explain what is meant by the term **sensor**.

....

....

....

....[2]

(b) State **two** sensors that could be used in this system and describe how they could be used.

Sensor 1 ....

Use ....

....

....

....

....

Sensor 2 ....

Use ....

....

....

....

....

[6]

10 **(a)** Describe what is meant by Transport Layer Security (TLS).

....

....

....

....

....

....[3]

(b) Name **three** different applications of TLS.

1 ....

2 ....

3 ....

[3]

11 Complete the paragraphs choosing the correct five terms from the list. Each term can only be used once:

- Ethics
- Freeware
- Free Software
- Hacking
- Malware
- Plagiarism
- Shareware
- Virus

Taking another person's work from the Internet and claiming it as your own is called .... It is possible to protect your work online with copyright.

One product that people may want to protect is software. .... does allow a person to share, copy and change software freely, but .... does not allow a person to do this legally. Software that has a licence allowing free use for a trial period is called .... The name given to this area of Computer Science is ....

[5]

Cambridge International Examinations
Cambridge International General Certificate of Secondary Education

CANDIDATE NAME

CENTRE NUMBER

CANDIDATE NUMBER

COMPUTER SCIENCE **0478/13**

Paper 1 Theory **October/November 2017**

1 hour 45 minutes

Candidates answer on the Question Paper.

No Additional Materials are required.

No calculators allowed.

READ THESE INSTRUCTIONS FIRST

Write your Centre number, candidate number and name in the spaces at the top of this page.
Write in dark blue or black pen.
You may use an HB pencil for any diagrams, graphs or rough working.
Do not use staples, paper clips, glue or correction fluid.
DO **NOT** WRITE IN ANY BARCODES.

Answer **all** questions.
No marks will be awarded for using brand names of software packages or hardware.

At the end of the examination, fasten all your work securely together.
The number of marks is given in brackets [ ] at the end of each question or part question.

The maximum number of marks is 75.

This document consists of **11** printed pages and **1** blank page.

DC (CE/SW) 152857

1 A washing machine has a small display screen built into it.

One use of the display screen is to show an error code when a problem has occurred with a washing cycle.

(a) State whether the display screen is an **input**, **output** or **storage device**.

....[1]

(b) The display screen shows a hexadecimal error code:

E04

This error code means that the water will not empty out of the washing machine.

Convert this error code to binary.

[3]

(c) State why hexadecimal is used to display the error code.

....

....

....

....[1]

(d) Identify **three** sensors that could be used in the washing machine.

State what each sensor could be used for.

Sensor 1 ....

Use ....

....

Sensor 2 ....

Use ....

....

Sensor 3 ....

Use ....

....

[6]

2 Data files are stored in different file formats.

Complete the table by providing a suitable file format for each file type. The first one has been done for you.

File type	File format
Pictures	.JPEG
Text	
Sound	
Video	

[3]

3 **(a)** An example of a Uniform Resource Locator (URL) is:

$$\underbrace{\textbf{http://}}_{\text{Part 1}}\textbf{www.}\underbrace{\textbf{cie.org.uk}}_{\text{Part 2}}\underbrace{\textbf{/index.htm}}_{\text{Part 3}}$$

Identify the **three** parts that make up this URL.

Part 1 ....

Part 2 ....

Part 3 ....

[3]

(b) Describe what is meant by an Internet Protocol (IP) address.

....

....

....

....

....

....

....

....[4]

4 **Six** components of a computer system and **six** descriptions are shown.

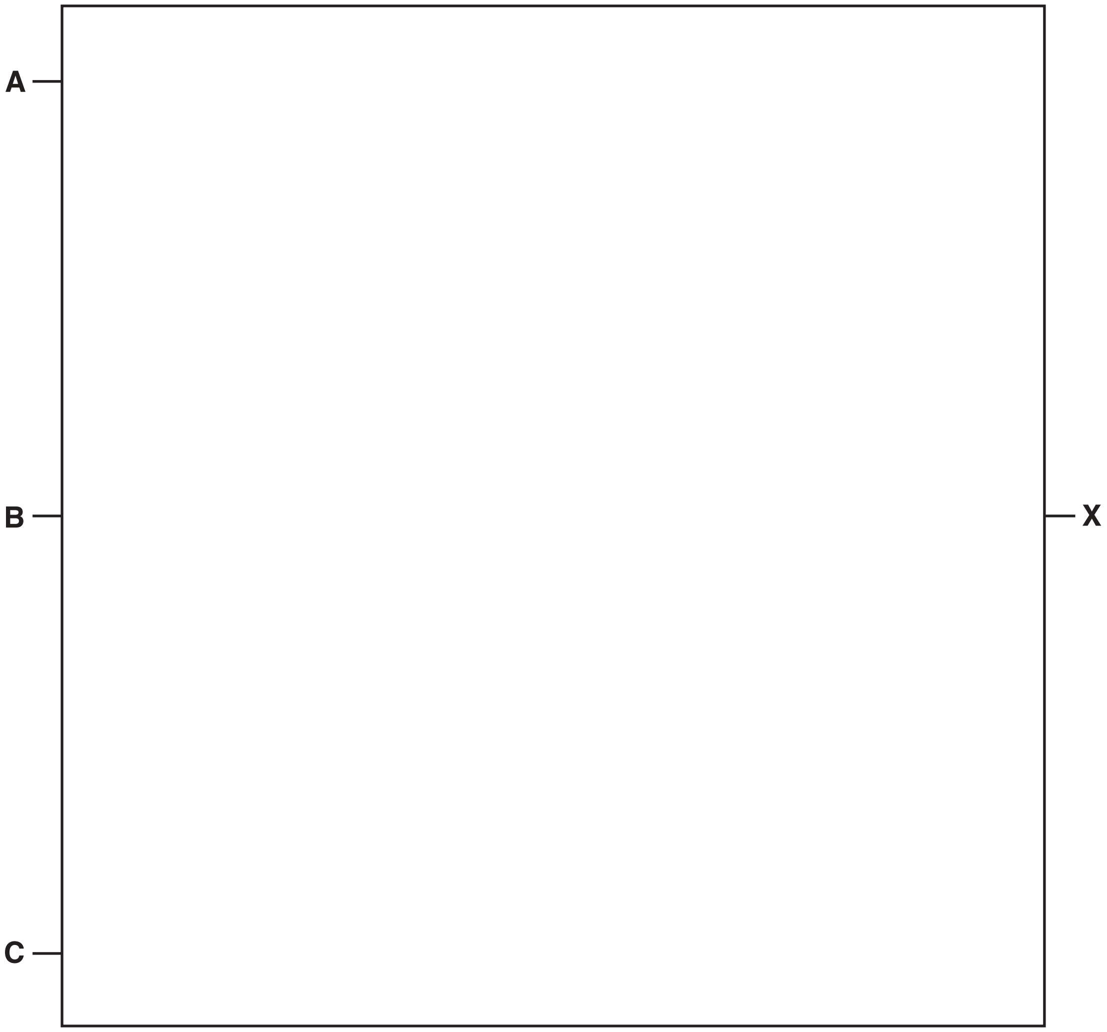
Draw a line to match each component with the most suitable description.

Component	**Description**
Arithmetic Logic Unit (ALU)	Used to connect together the internal components of the CPU.
Buses	Used to carry out calculations on data.
Control Unit (CU)	Used to temporarily hold data and instructions during processing.
Immediate Access Store (IAS)	Used to allow interaction with the computer.
Input/Output	Used to hold data and instructions before they are processed.
Registers	Used to manage the flow of data through the CPU.

[5]

5 **(a)** Draw a logic circuit for the logic statement:

X = 1 if ((A is 1 AND B is 1) OR (A is NOT 1 AND C is 1))



[4]

(b) Draw the symbol for an **XOR** gate and explain the function of this logic gate.

Explanation ....

....

....

....

....

....[5]

6 Describe the operation of a 2D scanner and a 3D scanner.

2D ....

....

....

....

....

....

3D ....

....

....

....

....

....

[6]

7 **Six** statements about firewalls are shown.

Tick (✓) to show whether each statement is **true** or **false**.

Statement	**true** (✓)	**false** (✓)
Firewalls can monitor incoming and outgoing traffic.		
Firewalls operate by checking traffic against a set of rules.		
Firewalls cannot block access to a certain website.		
Firewalls can be software and hardware.		
Firewalls can act as intermediary servers.		
Firewalls can block unauthorised traffic.		

[6]

8 **(a)** Data is valuable. It needs to be kept secure and it can easily be damaged.

Give **three** different ways that data can be accidentally damaged.

1 ....

....

2 ....

....

3 ....

....
[3]

(b) The Secure Socket Layer (SSL) protocol can be used to securely transmit data in online banking.

State **three** other different applications that use SSL.

Application 1 ....

Application 2 ....

Application 3 ....
[3]

(c) Online banking is increasing in popularity.

Online banking can be a risk as it can raise a number of security issues. SSL can be used as a security method to make online banking safer.

Identify and describe **three** other security methods that could be used to make online banking safer.

Security method 1 ....

....

....

....

Security method 2 ....

....

....

....

Security method 3 ....

....

....

....

[6]

9 **(a)** Optical storage media can be used to store data.

Describe how the data is read from a Compact Disc (CD).

....

....

....

....

....

....

....

....[4]

(b) Kamil wants to store a 16-bit colour image file. The image size is 1000 pixels.

Calculate the size of the file.

Give your answer in **bytes**. Show your working.

Working ....

....

....

....

Answer .... **bytes**

[2]

(c) Describe the differences between primary and secondary storage.

....

....

....

....

....

....

....

....[4]

10 **Six** statements about assembly language are shown.

Tick (✓) whether the statement is **true** or **false**.

Statement	**true** (✓)	**false** (✓)
Assembly language uses mnemonic codes.		
Assembly language programs do not need a translator to be executed.		
Assembly language is a low-level programming language.		
Assembly language is specific to the computer hardware.		
Assembly language is machine code.		
Assembly language is often used to create drivers for hardware.		

[6]

BLANK PAGE

Cambridge International Examinations
Cambridge International General Certificate of Secondary Education

CANDIDATE NAME	

CENTRE NUMBER						CANDIDATE NUMBER					

9777315973

COMPUTER SCIENCE **0478/21**

Paper 2 Problem-solving and Programming **October/November 2017**

1 hour 45 minutes

Candidates answer on the Question Paper.

No Additional Materials are required.

No calculators allowed.

READ THESE INSTRUCTIONS FIRST

Write your Centre number, candidate number and name in the spaces at the top of this page.
Write in dark blue or black pen.
You may use an HB pencil for any diagrams, graphs or rough working.
Do not use staples, paper clips, glue or correction fluid.
DO **NOT** WRITE IN ANY BARCODES.

Answer **all** questions.

DO NOT ATTEMPT TASKS 1, 2 AND 3 in the pre-release material; these are for information only.

You are advised to spend no more than **40 minutes** on **Section A** (Question 1).

No marks will be awarded for using brand names of software packages or hardware.

At the end of the examination, fasten all your work securely together.

The number of marks is given in brackets [ ] at the end of each question or part question.

The maximum number of marks is 50.

The syllabus is approved for use in England, Wales and Northern Ireland as a Cambridge International Level 1/Level 2 Certificate.

This document consists of **12** printed pages.

DC (CW/SW) 131978/2

Section A

You are advised to spend no longer than 40 minutes answering this section.

Here is a copy of the pre-release material.

DO NOT attempt Tasks 1, 2 and 3 now.

Use the pre-release material and your experience from attempting the tasks before the examination to answer Question 1.

Pre-release material

A small airfield operates a flying club where people can take a short flight to see if they would like flying lessons. The owner of the airfield has asked you to write a program to organise the flight bookings. The airfield operates three different planes and offers either a 30 minute or a 60 minute flight.

The following table shows the tariff:

Length of Flight	**2 Seater Plane**	**4 Seater Plane**	**Historic Plane**
30 minutes	$100	$120	$300
60 minutes	$150	$200	$500

After each flight, 30 minutes must be allowed for refuelling and safety checks before the next flight can take off. All planes offer both 30 minute and 60 minute flights, but, for the purpose of this activity, they will not be mixed on a given day, e.g. the 2 seater will offer 30 minute flights ONLY on one day and 60 minute flights ONLY on another day.

Write and test a program for the owner of the airfield.

- Your program must include appropriate prompts for the entry of data.
- Error messages and other output need to be set out clearly and understandably.
- All variables, constants and other identifiers must have meaningful names.

You will need to complete these **three** tasks. Each task must be fully tested.

TASK 1 – Work out the maximum income.
Assume that the flights take place between 08:00 and 18:00. Write a program that will work out the maximum income that can be generated by each plane in a day for each type of flight. The program should allow you to:

- input the type of plane
- input the length of flight
- calculate the maximum number of flights in a day
- output the total possible income per day for the choice of plane and length of flight

TASK 2 – Record bookings.
Write a program to store bookings for each plane and to allow you to find which planes are available at a given time slot during the day. The program should calculate the actual number of flights taken by each plane in that day.

TASK 3 – Work out income.
Modify TASK 2 so that it will calculate the total amount of money taken in a day for each plane, as well as the overall daily total for all three planes, and output the results.

1 **(a)** All variables, constants and other identifiers should have meaningful names.

(i) Declare suitable arrays for each of the planes to store time slots booked.

....

....

....[3]

(ii) Name **one** variable and **one** constant you used for **Task 1** and state the purpose of each one.

Variable ....

Purpose ....

....

Constant ....

Purpose ....

....

[4]

[Turn over

(b) Write an algorithm to complete **Task 1**, using **either** pseudocode, programming statements or a flowchart.

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....[5]

(c) Describe how you could validate and test the inputs for **Task 1**.

....

....

....

....

....

....

....

....[4]

(d) Explain how your program checks and displays whether any of the planes are available at a given time of the day (**Task 2**). Any programming statements used must be fully explained.

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....[4]

Section B

2 This section of program code asks for 80 numbers between 100 and 1000 to be entered. It checks that the numbers are in the correct range, and stores them in an array. It counts how many of the numbers are larger than 500 and then outputs the result when the program is finished.

```
1 Count = 0
2 FOR Index = 1 TO 80
3   INPUT 'Enter a number between 100 and 1000', Number
4   WHILE Number = 99 AND Number = 1001
5     INPUT 'This is incorrect, please try again', Number
6   ENDWHILE
7   Num[80] = Number
8   IF Number > 500 THEN Count = Count + 1
9  UNTIL Index = 80
10 PRINT Index
11 PRINT ' numbers were larger than 500'
```

There are **four** lines of code that contain errors.

State the line number for each error and write the correct code for that line.

Error 1 Line Number ....

Correct Code ....

Error 2 Line Number ....

Correct Code ....

Error 3 Line Number ....

Correct Code ....

Error 4 Line Number ....

Correct Code ....

[4]

3 **(a)** Explain the difference between a validation check and a verification check.

....

....

....[2]

(b) Describe, using an example, how data could be verified on data entry.

....

....

....[2]

(c) Explain what is meant by the term library routine.

....

....

....[2]

4 **(a)** **Four** pseudocode descriptions and **five** pseudocode statements are shown. Draw one line to link each pseudocode description to the correct pseudocode statement. Not all pseudocode statements will be used.

Pseudocode description	**Pseudocode statement**
A loop that will iterate at least once.	`FOR…TO…NEXT`
A conditional statement to deal with many possible outcomes.	`IF…THEN…ELSE…ENDIF`
A loop that will iterate a set number of times.	`WHILE…DO…ENDWHILE`
A conditional statement with different outcomes for true and false.	`CASE…OF…OTHERWISE…ENDCASE`
	`REPEAT…UNTIL`

[4]

(b) Write an algorithm in pseudocode, using a single loop, to print 50 names that have been stored in an array.

....

....

....

....

....

....

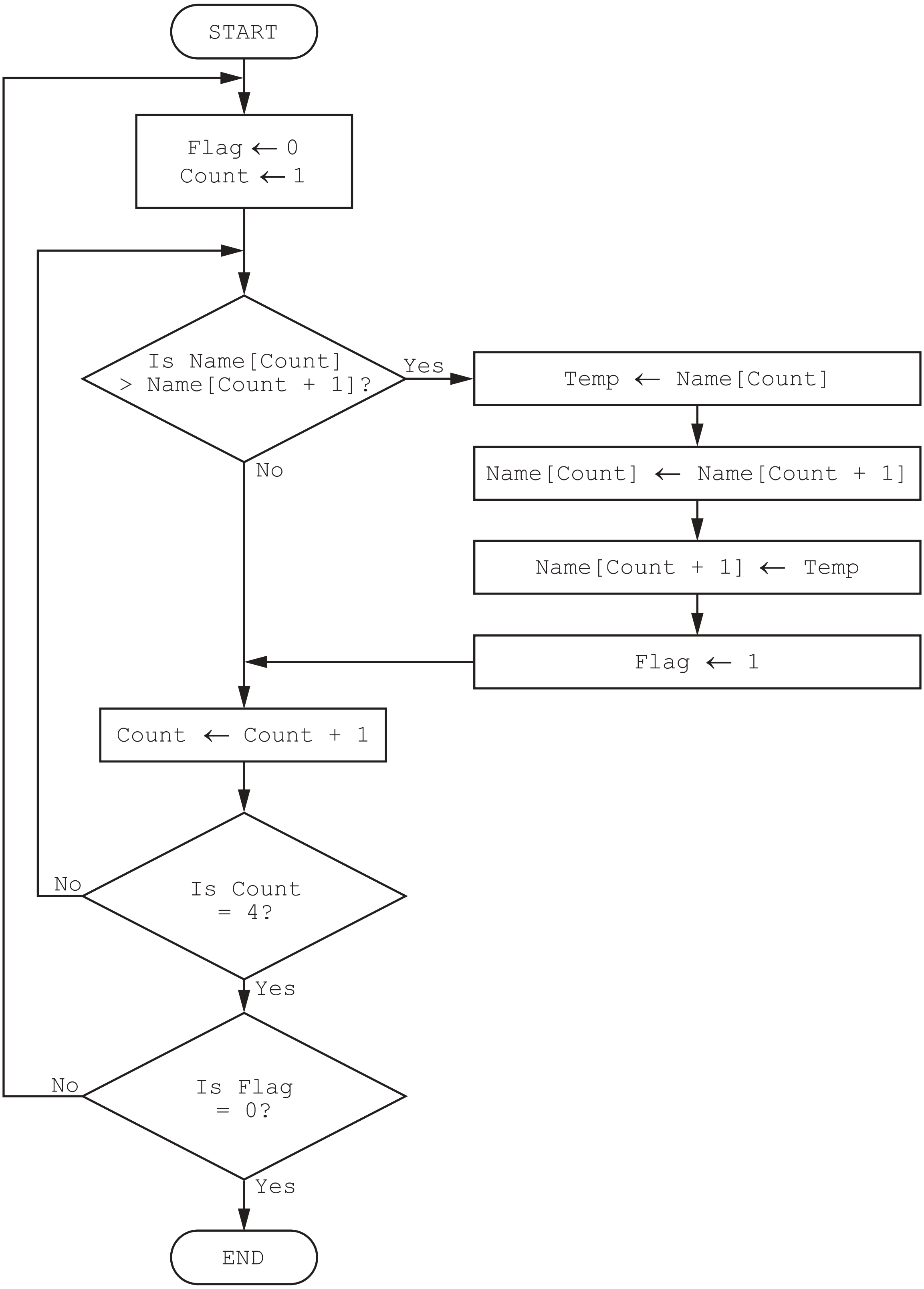
....

....

....

....[3]

5 The flowchart below represents a program routine.



(a) The array used in the flowchart contains the following data:

Name[1]	**Name[2]**	**Name[3]**	**Name[4]**
Jamal	Amir	Eve	Tara

Complete the trace table using the data given in the array.

Flag	**Count**	**Name[1]**	**Name[2]**	**Name[3]**	**Name[4]**	**Temp**
		Jamal	Amir	Eve	Tara	

[5]

(b) Describe what the algorithm represented by the flowchart is doing.

....

....[2]

Question 6 begins on Page 12.

6 A wildlife park has a database table, called LIVESTOCK, to classify and record its animal species. Part of the database table is shown.

Species	Classification	Diet	Legs
Giraffe	Mammal	Herbivore	4
Elephant	Mammal	Herbivore	4
Crocodile	Reptile	Carnivore	4
Ostrich	Bird	Omnivore	2
Gorilla	Mammal	Herbivore	2
Bear	Mammal	Omnivore	4
Rhinoceros	Mammal	Herbivore	4
Hippopotamus	Mammal	Herbivore	4
Flamingo	Bird	Omnivore	2
Lion	Mammal	Carnivore	4
Turtle	Reptile	Omnivore	4
Penguin	Bird	Carnivore	2

(a) Suggest another appropriate field that could be added to this database by stating its name and data type. State its purpose and give an example of the data it could contain.

Field name ....

Data Type ....

Purpose ....

....

Example of data ....

[2]

(b) Use the query-by-example grid below to provide a list of all four legged mammals that are herbivores, sorted alphabetically by species, with only the species displayed.

Field:					
Table:					
Sort:					
Show:	☐	☐	☐	☐	☐
Criteria:					
or:					

[4]

Cambridge International Examinations
Cambridge International General Certificate of Secondary Education

CANDIDATE NAME	

CENTRE NUMBER		CANDIDATE NUMBER	

COMPUTER SCIENCE **0478/22**

Paper 2 Problem-solving and Programming **October/November 2017**

1 hour 45 minutes

Candidates answer on the Question Paper.

No Additional Materials are required.

No calculators allowed.

READ THESE INSTRUCTIONS FIRST

Write your Centre number, candidate number and name in the spaces at the top of this page.
Write in dark blue or black pen.
You may use an HB pencil for any diagrams, graphs or rough working.
Do not use staples, paper clips, glue or correction fluid.
DO **NOT** WRITE IN ANY BARCODES.

Answer **all** questions.

DO NOT ATTEMPT TASKS 1, 2 AND 3 in the pre-release material; these are for information only.

You are advised to spend no more than **40 minutes** on **Section A** (Question 1).

No marks will be awarded for using brand names of software packages or hardware.

At the end of the examination, fasten all your work securely together.

The number of marks is given in brackets [ ] at the end of each question or part question.

The maximum number of marks is 50.

This document consists of **10** printed pages and **2** blank pages.

DC (CW/FD) 132152/3

Section A

You are advised to spend no longer than 40 minutes answering this section.

Here is a copy of the pre-release material.

DO NOT attempt Tasks 1, 2 and 3 now.

Use the pre-release material and your experience from attempting the tasks before the examination to answer Question 1.

Pre-release material

The owner of a river boat hire company wants to calculate the daily profits from hiring out 10 rowing boats on the river. Boats are numbered 1 to 10. Boats can be hired for use between 10:00 and 17:00 every day.

Write and test a program for the owner.

- Your program must include appropriate prompts for the entry of data.
- Error messages and other output need to be set out clearly and understandably.
- All variables, constants and other identifiers must have meaningful names.

You will need to complete these **three** tasks. Each task must be fully tested.

TASK 1 – calculate the money taken in a day for one boat.

The cost of hiring a boat is $20 for one hour or $12 for half an hour. When a boat is hired the payment is added to the money taken for the day. The running total of hours hired that day is updated and the time when the boat must be returned is stored. At the end of the day the money taken and the total hours hired is output.
No boat can be hired before 10:00 or returned after 17:00.

TASK 2 – find the next boat available.

Extend TASK 1 to work for all 10 rowing boats. Use the data stored for each boat to find out how many boats are available for hire at the current time. If no boats are available show the earliest time that a boat will be available for hire.

TASK 3 – calculate the money taken for all the boats at the end of the day.

At the end of the day use the data stored for each boat to calculate the total amount of money taken and the total number of hours boats were hired that day. Find out how many boats were not used that day and which boat was used the most. Provide a report for the owner to show this information.

1 **(a)** All variables, constants and other identifiers should have meaningful names.

(i) For **one** variable that you have used to record the information about a single boat in **Task 1**, state the name, data type and its use.

Variable name ....

Data type ....

Use .... [3]

(ii) State **one** constant and its value that you could have used for **Task 1**.

Constant name ....

Value .... [2]

(b) Give **two** different validation checks you could have used for data entry in **Task 1**. For each check explain why it could be used and provide a set of data for testing.

Validation check 1 ....

....

Reason for choice ....

....

Set of test data ....

....

Validation check 2 ....

....

Reason for choice ....

....

Set of test data ....

.... [6]

(c) Write an algorithm to complete **Task 3**, using **either** pseudocode, programming statements **or** a flowchart. You may assume Task 2 has been completed.

[5]

(d) Explain how your program finds out how many boats are available for hire (**Task 2**).
Any programming statements used must be fully explained.

....[4]

Section B

2 Write an algorithm using **either** pseudocode **or** a flowchart, to:

- input a positive integer
- use this value to set up how many other numbers are to be input
- input these numbers
- calculate and output the total and the average of these numbers.

[6]

3 The following diagram shows **four** data structures and **four** descriptions.

Draw a line to connect each data structure to the correct description.

Data structure	**Description**
Constant	A collection of related data
Array	A value that can change whilst a program is running
Table	A value that never changes whilst a program is running
Variable	A series of elements of the same data type

[3]

4 `IF … THEN … ELSE … ENDIF` is one type of conditional statement used when writing pseudocode.

Identify and describe **another** type of conditional statement that you could use when writing pseudocode. Give a reason why you would use this type of conditional statement.

Conditional statement ....

....

....

....

Description ....

....

....

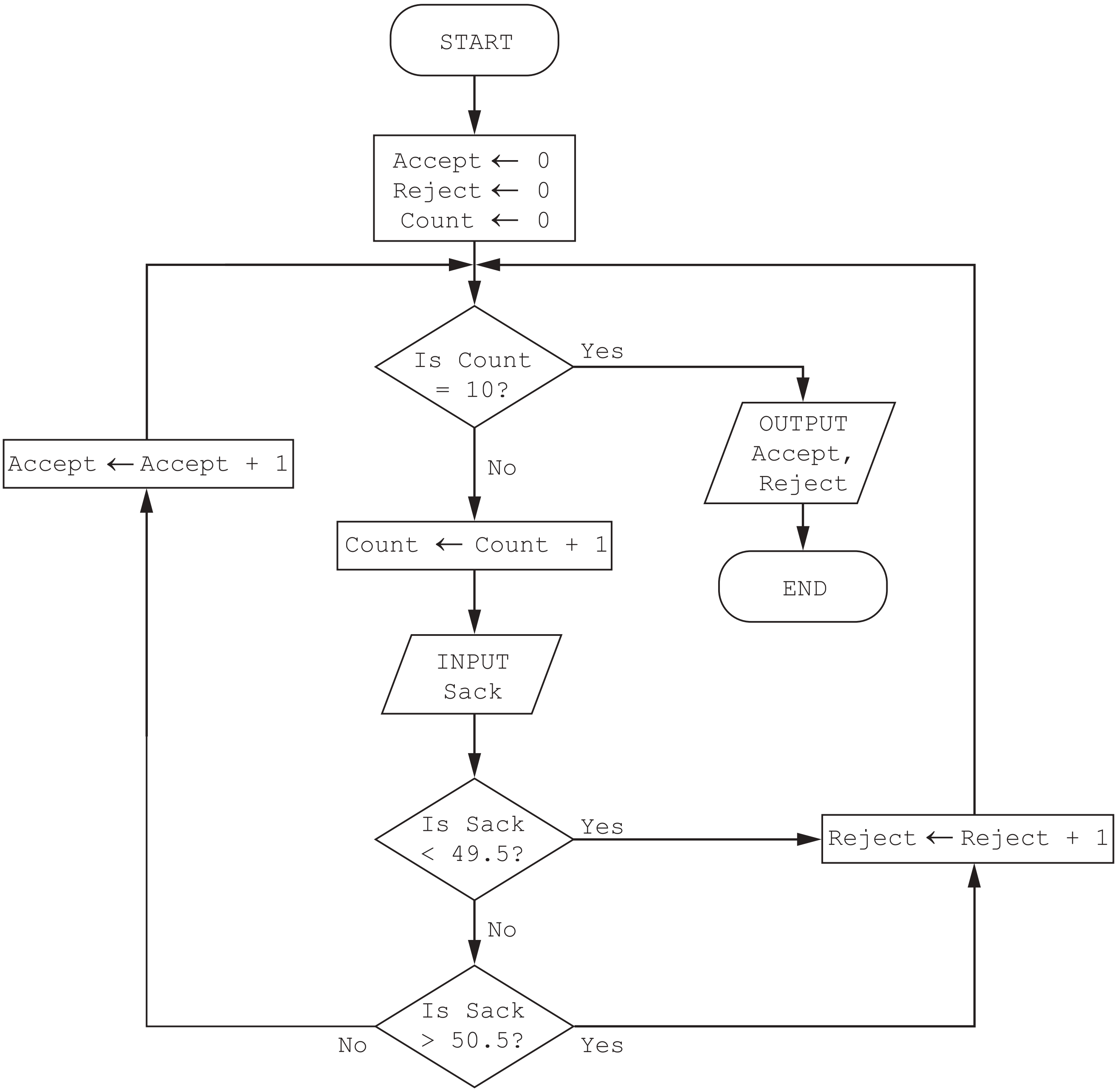
....

Reason ....

....

[4]

5 **(a)** This flowchart checks a batch of 10 rice sacks for weight. Sacks should weigh 50 kilograms each. Sacks weighing over 50.5 kilograms or less than 49.5 kilograms are rejected. The number of sacks accepted and the number of sacks rejected is output.



Complete the trace table for the input data:

50.4, 50.3, 49.1, 50.3, 50.0, 49.5, 50.2, 50.3, 50.5, 50.6

Accept	Reject	Count	Sack	OUTPUT

[5]

(b) The size of the batch has increased to 50 sacks. It has been decided to only reject sacks that are underweight.

State the changes that need to be made to the flowchart.

....

....

....

....[2]

6 A database table, TRAIN, is to be set up for a railway company to keep a record of the engines available for use. Each engine has a unique number made up of 5 digits, nnnnn. The engines are classified as freight (F) or passenger (P) together with a power classification that is a whole number between 0 and 9, for example F8. The railway company keeps a record of the date of the last service for each engine.

(a) Identify the **three** fields required for the database. Give each field a suitable name and data type. Provide a sample of data that you could expect to see in the field.

Field 1 Name ....

Data type ....

Data sample ....

Field 2 Name ....

Data type ....

Data sample ....

Field 3 Name ....

Data type ....

Data sample ....[6]

(b) State the field that you should choose as the primary key.

....[1]

(c) Using the query-by-example grid below, write a query to identify all passenger engines that have not been serviced in the past 12 months. Only display the engine numbers.

Field:				
Table:				
Sort:				
Show:	☐	☐	☐	☐
Criteria:				
or:				

[3]

BLANK PAGE

BLANK PAGE

Cambridge International Examinations
Cambridge International General Certificate of Secondary Education

CANDIDATE NAME

CENTRE NUMBER

CANDIDATE NUMBER

COMPUTER SCIENCE **0478/23**

Paper 2 Problem-solving and Programming **October/November 2017**

1 hour 45 minutes

Candidates answer on the Question Paper.

No Additional Materials are required.

No calculators allowed.

READ THESE INSTRUCTIONS FIRST

Write your Centre number, candidate number and name in the spaces at the top of this page.
Write in dark blue or black pen.
You may use an HB pencil for any diagrams, graphs or rough working.
Do not use staples, paper clips, glue or correction fluid.
DO **NOT** WRITE IN ANY BARCODES.

Answer **all** questions.

DO NOT ATTEMPT TASKS 1, 2 AND 3 in the pre-release material; these are for information only.

You are advised to spend no more than **40 minutes** on **Section A** (Question 1).

No marks will be awarded for using brand names of software packages or hardware.

At the end of the examination, fasten all your work securely together.

The number of marks is given in brackets [ ] at the end of each question or part question.

The maximum number of marks is 50.

This document consists of **12** printed pages.

DC (LK) 150377

Section A

You are advised to spend no longer than 40 minutes answering this section.

Here is a copy of the pre-release material.

DO NOT attempt Tasks 1, 2 and 3 now.

Use the pre-release material and your experience from attempting the tasks before the examination to answer Question 1.

Pre-release material

A small airfield operates a flying club where people can take a short flight to see if they would like flying lessons. The owner of the airfield has asked you to write a program to organise the flight bookings. The airfield operates three different planes and offers either a 30 minute or a 60 minute flight.

The following table shows the tariff:

Length of Flight	2 Seater Plane	4 Seater Plane	Historic Plane
30 minutes	\$100	\$120	\$300
60 minutes	\$150	\$200	\$500

After each flight, 30 minutes must be allowed for refuelling and safety checks before the next flight can take off. All planes offer both 30 minute and 60 minute flights, but, for the purpose of this activity, they will not be mixed on a given day, e.g. the 2 seater will offer 30 minute flights ONLY on one day and 60 minute flights ONLY on another day.

Write and test a program for the owner of the airfield.

- Your program must include appropriate prompts for the entry of data.
- Error messages and other output need to be set out clearly and understandably.
- All variables, constants and other identifiers must have meaningful names.

You will need to complete these **three** tasks. Each task must be fully tested.

TASK 1 – Work out the maximum income.
Assume that the flights take place between 08:00 and 18:00. Write a program that will work out the maximum income that can be generated by each plane in a day for each type of flight. The program should allow you to:

- input the type of plane
- input the length of flight
- calculate the maximum number of flights in a day
- output the total possible income per day for the choice of plane and length of flight

TASK 2 – Record bookings.
Write a program to store bookings for each plane and to allow you to find which planes are available at a given time slot during the day. The program should calculate the actual number of flights taken by each plane in that day.

TASK 3 – Work out income.
Modify TASK 2 so that it will calculate the total amount of money taken in a day for each plane, as well as the overall daily total for all three planes, and output the results.

1 **(a)** All variables, constants and other identifiers should have meaningful names.

(i) Declare suitable arrays for each of the planes to store time slots booked.

....

....

....[3]

(ii) Name **one** variable and **one** constant you used for **Task 1** and state the purpose of each one.

Variable ....

Purpose ....

....

Constant ....

Purpose ....

....
[4]

(b) Write an algorithm to complete **Task 1**, using **either** pseudocode, programming statements or a flowchart.

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....[5]

(c) Describe how you could validate and test the inputs for **Task 1**.

....

....

....

....

....

....

....

....[4]

(d) Explain how your program checks and displays whether any of the planes are available at a given time of the day (**Task 2**). Any programming statements used must be fully explained.

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....[4]

Section B

2 This section of program code asks for 80 numbers between 100 and 1000 to be entered. It checks that the numbers are in the correct range, and stores them in an array. It counts how many of the numbers are larger than 500 and then outputs the result when the program is finished.

```
1 Count = 0
2 FOR Index = 1 TO 80
3   INPUT 'Enter a number between 100 and 1000', Number
4   WHILE Number = 99 AND Number = 1001
5     INPUT 'This is incorrect, please try again', Number
6   ENDWHILE
7   Num[80] = Number
8   IF Number > 500 THEN Count = Count + 1
9  UNTIL Index = 80
10 PRINT Index
11 PRINT ' numbers were larger than 500'
```

There are **four** lines of code that contain errors.

State the line number for each error and write the correct code for that line.

Error 1 Line Number ....

Correct Code ....

Error 2 Line Number ....

Correct Code ....

Error 3 Line Number ....

Correct Code ....

Error 4 Line Number ....

Correct Code ....

[4]

3 **(a)** Explain the difference between a validation check and a verification check.

....

....

....[2]

(b) Describe, using an example, how data could be verified on data entry.

....

....

....[2]

(c) Explain what is meant by the term library routine.

....

....

....[2]

4 **(a)** **Four** pseudocode descriptions and **five** pseudocode statements are shown. Draw one line to link each pseudocode description to the correct pseudocode statement. Not all pseudocode statements will be used.

Pseudocode description

- A loop that will iterate at least once.
- A conditional statement to deal with many possible outcomes.
- A loop that will iterate a set number of times.
- A conditional statement with different outcomes for true and false.

Pseudocode statement

- `FOR…TO…NEXT`
- `IF…THEN…ELSE…ENDIF`
- `WHILE…DO…ENDWHILE`
- `CASE…OF…OTHERWISE…ENDCASE`
- `REPEAT…UNTIL`

[4]

(b) Write an algorithm in pseudocode, using a single loop, to print 50 names that have been stored in an array.

....

....

....

....

....

....

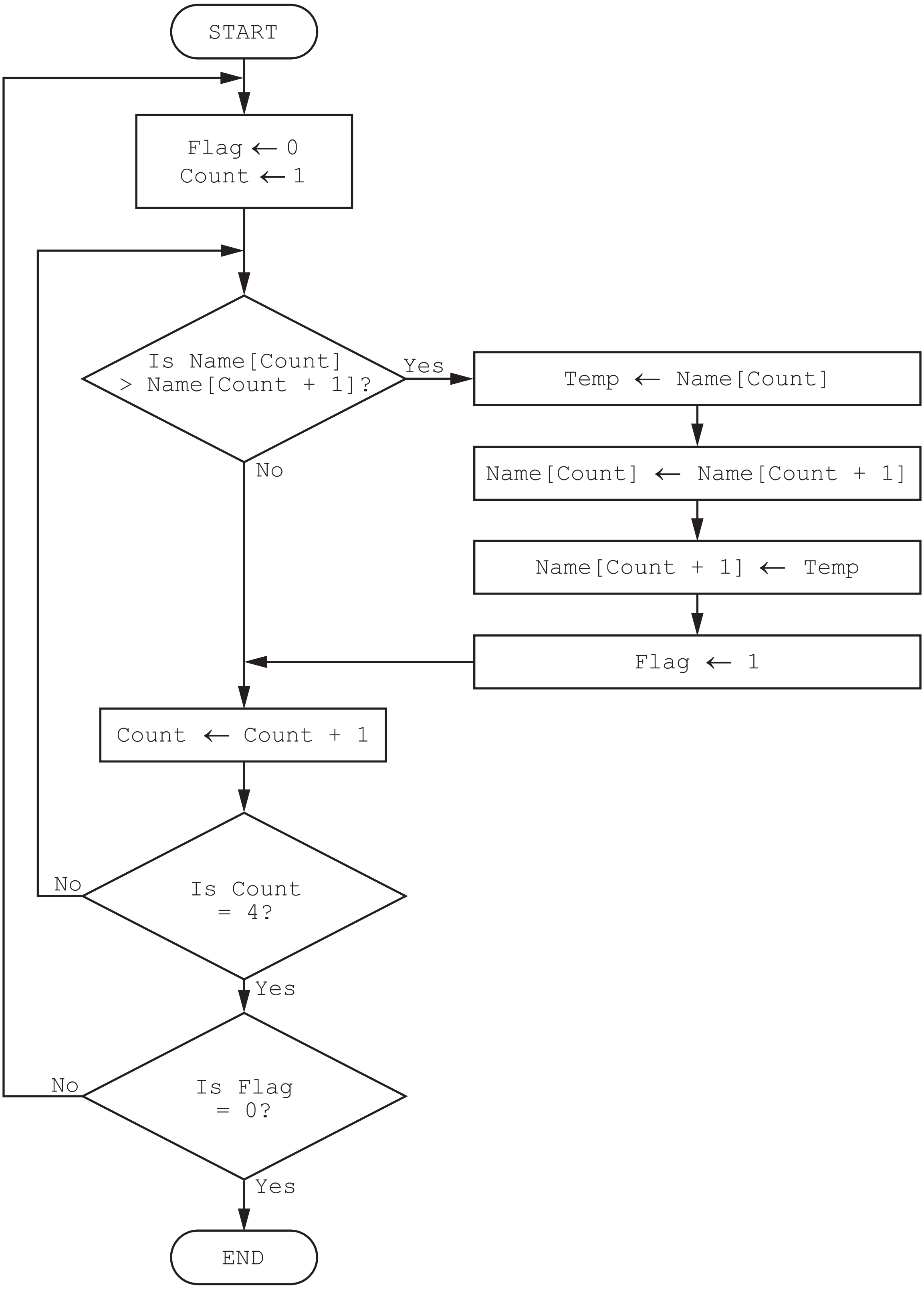
....

....

....

....[3]

5 The flowchart below represents a program routine.



(a) The array used in the flowchart contains the following data:

Name[1]	**Name[2]**	**Name[3]**	**Name[4]**
Jamal	Amir	Eve	Tara

Complete the trace table using the data given in the array.

Flag	**Count**	**Name[1]**	**Name[2]**	**Name[3]**	**Name[4]**	**Temp**
		Jamal	Amir	Eve	Tara	

[5]

(b) Describe what the algorithm represented by the flowchart is doing.

....

....[2]

Question 6 begins on Page 12.

6 A wildlife park has a database table, called LIVESTOCK, to classify and record its animal species. Part of the database table is shown.

Species	Classification	Diet	Legs
Giraffe	Mammal	Herbivore	4
Elephant	Mammal	Herbivore	4
Crocodile	Reptile	Carnivore	4
Ostrich	Bird	Omnivore	2
Gorilla	Mammal	Herbivore	2
Bear	Mammal	Omnivore	4
Rhinoceros	Mammal	Herbivore	4
Hippopotamus	Mammal	Herbivore	4
Flamingo	Bird	Omnivore	2
Lion	Mammal	Carnivore	4
Turtle	Reptile	Omnivore	4
Penguin	Bird	Carnivore	2

(a) Suggest another appropriate field that could be added to this database by stating its name and data type. State its purpose and give an example of the data it could contain.

Field name ....

Data Type ....

Purpose ....

....

Example of data .... [2]

(b) Use the query-by-example grid below to provide a list of all four legged mammals that are herbivores, sorted alphabetically by species, with only the species displayed.

Field:					
Table:					
Sort:					
Show:	☐	☐	☐	☐	☐
Criteria:					
or:					

[4]

Cambridge Assessment
International Education

Grade thresholds – March 2018

Cambridge IGCSE Computer Science (0478)

Grade thresholds taken for Syllabus 0478 (Computer Science) in the March 2018 examination.

		minimum raw mark required for grade:						
	maximum raw mark available	A	B	C	D	E	F	G
Component 12	75	53	45	37	32	28	23	18
Component 22	50	36	29	22	18	14	11	8

Grade A* does not exist at the level of an individual component.

The maximum total mark for this syllabus, after weighting has been applied, is **125**.

The overall thresholds for the different grades were set as follows.

Option	Combination of Components	A*	A	B	C	D	E	F	G
AY	12, 22	104	89	74	59	50	42	34	26

Learn more! For more information please visit **www.cambridgeinternational.org/igcse** or contact Customer Services on **+44 (0)1223 553554** or email **info@cambridgeinternational.org**

Cambridge Assessment International Education
Cambridge International General Certificate of Secondary Education

---

COMPUTER SCIENCE **0478/12**

Paper 1 **March 2018**

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 March 2018 series for most Cambridge IGCSE®, Cambridge International A and AS Level components and some Cambridge O Level components.

This syllabus is approved for use in England, Wales and Northern Ireland as a Cambridge International Level 1/Level 2 Certificate.

This document consists of **11** printed pages.

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:

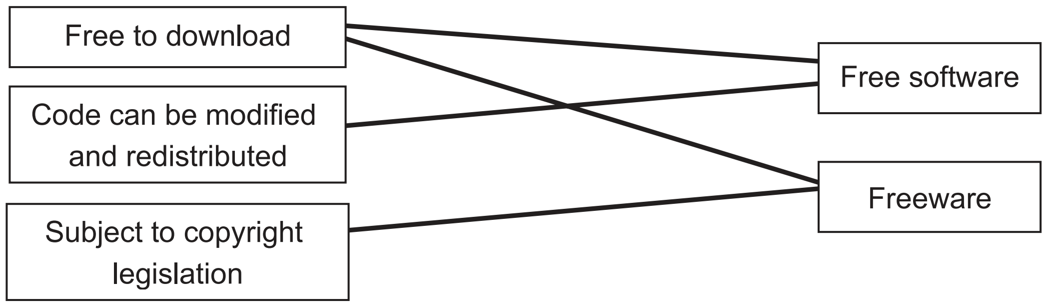
Rules must be applied consistently e.g. in situations where candidates have not followed instructions or in the application of generic level descriptors.

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	**Description** **Type of software**  1 mark for each 'Type of software' if correct lines are attached to it.	**2**

Question	Answer	Marks
2(a)	Any **three** from: <u>Scans</u> files for viruses // detects/identifies a virus Can constantly run in background Can run a scheduled scan Can automatically updating virus definitions Can quarantine a virus Can delete a virus Completes heuristic checking Notifies user of a possible virus	**3**

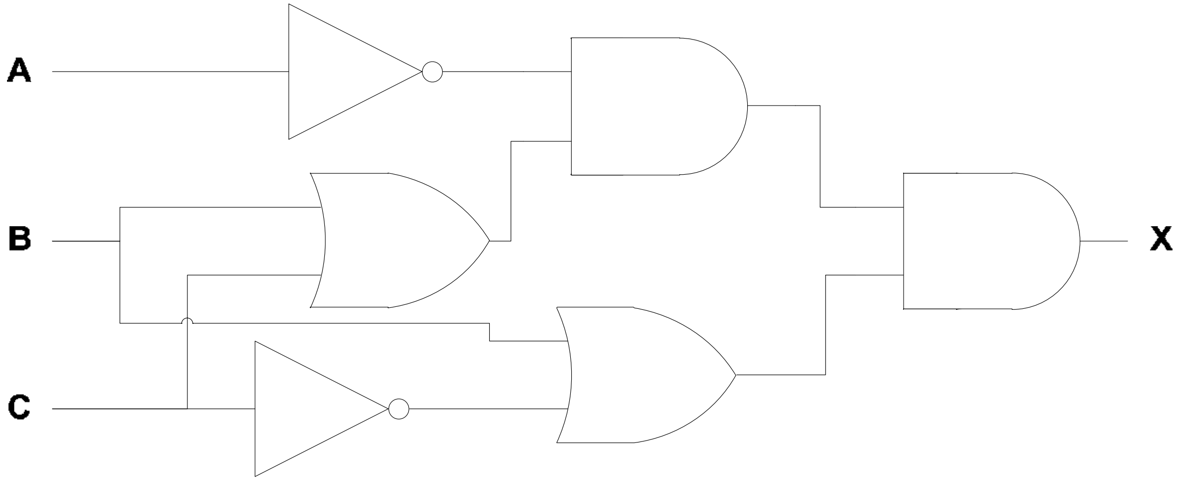
Question	Answer	Marks
2(b)	Any **three** from: Use a firewall Use of a proxy server Do not use / download software / files from unknown sources Do not share external storage devices / USB pens Do not open / take care when opening attachments / link Do not connect computer to network / use as stand-alone computer Limiting access to the computer	**3**

Question	Answer	Marks
3(a)	Byte 3 / 10110100	**1**
3(b)	Odd parity used Counted / added the number 1’s // Most Bytes have an odd number of 1’s Byte 3 has an even number of 1’s // Byte 3 didn’t follow odd parity	**3**

Question	Answer	Marks
4	Any **six** from: Sensor(s) send data/signals to the microprocessor Analogue signal/data from sensor is converted to digital (using ADC) Microprocessor compares data value against set boundaries / pre-set data If value between 21 and 24 … … no action taken If value > 24 °C / signal is sent from microprocessor… … to turn conditioning unit ON//Set to cold If value is < 21 °C signal is sent from microprocessor… … to turn conditioning unit ON//Set to warm Process is repeated for a continuous operation	**6**

Question	Answer	Marks
5(a)	One mark for each correct Hexadecimal value C4 | 10 | FE | 09	**3**
5(b)	Any **two** from: Easier / simpler to remember / write down // quicker to transcribe Less likely to make error Less digits to use	**2**

Question	Answer	Marks
6	1 mark for suitable example and 1 mark for suitable associated use. Primary RAM … … stores OS when running / data currently in use / instructions currently in use ROM … … stores boot instructions / BIOS Cache … … stores frequently used instructions Secondary HDD / SSD… …stores files / applications (by example) Offline CD/DVD/Blu-ray/Flash Memory/USB stick/Removable HDD/Removable SSD… … stores files / applications (by example)	**6**

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>7(a)</td><td>4 marks for 8 correct outputs
3 marks for 6 correct outputs
2 marks for 4 correct outputs
1 mark for 2 correct outputs
<table>
<tr><th>A</th><th>B</th><th>C</th><th>X</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>1</td></tr>
</table></td><td>4</td></tr>
<tr><td>7(b)</td><td>1 mark per gate in correct location

</td><td>6</td></tr>
</table>

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>7(c)</td><td>4 marks for 8 correct outputs
3 marks for 6 correct outputs
2 marks for 4 correct outputs
1 mark for 2 correct outputs
<table>
<tr><th>A</th><th>B</th><th>C</th><th>X</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>0</td></tr>
</table>
</td><td>4</td></tr>
</table>

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>8(a)</td><td>
<table>
<tr><th rowspan="2">Descriptions</th><th colspan="2">Method</th><th></th></tr>
<tr><td>Serial</td><td>Parallel</td><td></td></tr>
<tr><td>Multiple bits are sent and received at the same time.</td><td></td><td>✓</td><td>[1]</td></tr>
<tr><td>Bits are sent one at a time in a single direction.</td><td>✓</td><td></td><td>[1]</td></tr>
<tr><td>Bits are sent using a single wire. Data can be sent or received, but not at the same time.</td><td>✓</td><td></td><td>[1]</td></tr>
</table>
</td><td>3</td></tr>
</table>

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>8(b)</td><td>
<table>
<tr><th rowspan="2">Descriptions</th><th colspan="3">Type</th><th></th></tr>
<tr><th>Simplex</th><th>Half-duplex</th><th>Duplex</th><th></th></tr>
<tr><td>Multiple bits are sent and received at the same time.</td><td></td><td></td><td>✓</td><td>[1]</td></tr>
<tr><td>Bits are sent one at a time in a single direction.</td><td>✓</td><td></td><td></td><td>[1]</td></tr>
<tr><td>Bits are sent using a single wire. Data can be sent or received, but not at the same time.</td><td></td><td>✓</td><td></td><td>[1]</td></tr>
</table>
</td><td>3</td></tr>
</table>

Question	Answer	Marks
9	Max 3 – 1 mark for correct answer and 2 marks for correct calculations. Any **two** from: 16000 × 32 512000 / 1024 Or 16000 × 8 128000 × 32 4096000 / 8 512000 / 1024 **Correct answer:** 500 kB	**3**

Question	Answer	Marks
10(a)(i)	10010	**1**
10(a)(ii)	11110001	**1**
10(b)	Any **four** from: The program is stored on a secondary storage device Data and instructions are moved to memory / RAM Data and instructions are stored in the same memory / RAM Data and instructions are moved to registers to be executed Instructions are fetched one at a time	**4**

Question	Answer	Marks
11(a)	Smaller file size reduces download / display time // reduces upload time	**1**
11(b)	Any **four** from: A compression algorithm is used Permanently deleting some data // file cannot be restored to original Colour depth / palette can be reduced Resolution can be reduced // number of pixels can be reduced Less bits will be required for each pixel / colour	**4**

Question	Answer	Marks
12(a)	Quicker to scan … … rather than type into a system Fewer errors … … no human input	**4**

Question	Answer	Marks
12(b)	Any **four** from: Uses a barcode reader / scanner Reader shines light / red laser at barcode White lines reflect (more) light Sensors / photoelectric cells detect light reflected back Different reflections / bars will convert to different binary values	**4**

Question	Answer	Marks
13	Any **four** from • (Provides an) interface • Loads / opens / installs / closes software • Manages the hardware // manages peripherals // spooling • Manages the transfer of programs into and out of memory • Divides processing time // processor management • Manages file handling • Manages error handling / interrupts • Manages security software • Manages utility software • Manages user accounts • Multitasking // Multiprocessing // Multiprogramming // Time slicing • Batch processing // real time processing	**4**

Cambridge IGCSE

Cambridge International Examinations
Cambridge International General Certificate of Secondary Education

CANDIDATE NAME			
CENTRE NUMBER		CANDIDATE NUMBER	

COMPUTER SCIENCE **0478/12**

Paper 1 Theory **February/March 2018**

1 hour 45 minutes

Candidates answer on the Question Paper.

No Additional Materials are required.

No calculators allowed.

READ THESE INSTRUCTIONS FIRST

Write your Centre number, candidate number and name in the spaces at the top of this page.
Write in dark blue or black pen.
You may use an HB pencil for any diagrams, graphs or rough working.
Do not use staples, paper clips, glue or correction fluid.
DO **NOT** WRITE IN ANY BARCODES.

Answer **all** questions.
No marks will be awarded for using brand names of software packages or hardware.

At the end of the examination, fasten all your work securely together.
The number of marks is given in brackets [ ] at the end of each question or part question.

The maximum number of marks is 75.

This syllabus is approved for use in England, Wales and Northern Ireland as a Cambridge International Level 1/Level 2 Certificate.

This document consists of **12** printed pages.

DC (LK/CGW) 161583/3 **R**

CAMBRIDGE
International Examinations

[Turn over

1 Some types of software can be described as free software or freeware.

Draw lines to link each description to a correct type of software. A description can be linked to more than one type of software.

Description

Free to download

Code can be modified and redistributed

Subject to copyright legislation

Type of software

Free software

Freeware

[2]

2 David has installed anti-virus software on his computer.

(a) State **three** tasks carried out by anti-virus software.

Task 1 ....

....

Task 2 ....

....

Task 3 ....

....

[3]

(b) David is still concerned that his computer might get infected by a computer virus.

State **three** other ways in which David can reduce the risk of his computer getting a computer virus.

1 ....

....

2 ....

....

3 ....

....
[3]

3 Parity checks can be used to check for errors during data transmission.

One of the bytes has been transmitted incorrectly.

Byte 1	Byte 2	Byte 3	Byte 4
10110011	10101000	10110100	10110101

(a) State which byte was incorrectly transmitted.

....[1]

(b) Explain how you identified the incorrectly transmitted byte.

....

....

....

....

....

....[3]

4 An air conditioning system is used to control the temperature in a hospital.

The air conditioning system uses temperature sensors and a microprocessor.

The temperature must remain between 21 °C and 24 °C.

Describe how the sensors and the microprocessor are used to control the temperature of the hospital.

....

....

....

....

....

....

....

....

....

....[6]

5 The IP address of a computer is stored as a set of four 8-bit binary numbers.

The network administrator converts each binary number into hexadecimal.

(a) Complete the table to show the hexadecimal equivalent of the binary IP address.
The first number has already been converted.

Binary IP address

11000100	00010000	11111110	00001001

Hexadecimal

C4			

[3]

(b) Explain why the network administrator uses hexadecimal.

....

....

....

....[2]

6 Primary, secondary and off-line are types of storage.

Give an example of each type of storage.

For each example state how it is used.

Primary storage

Example ....

Use ....

....

Secondary storage

Example ....

Use ....

....

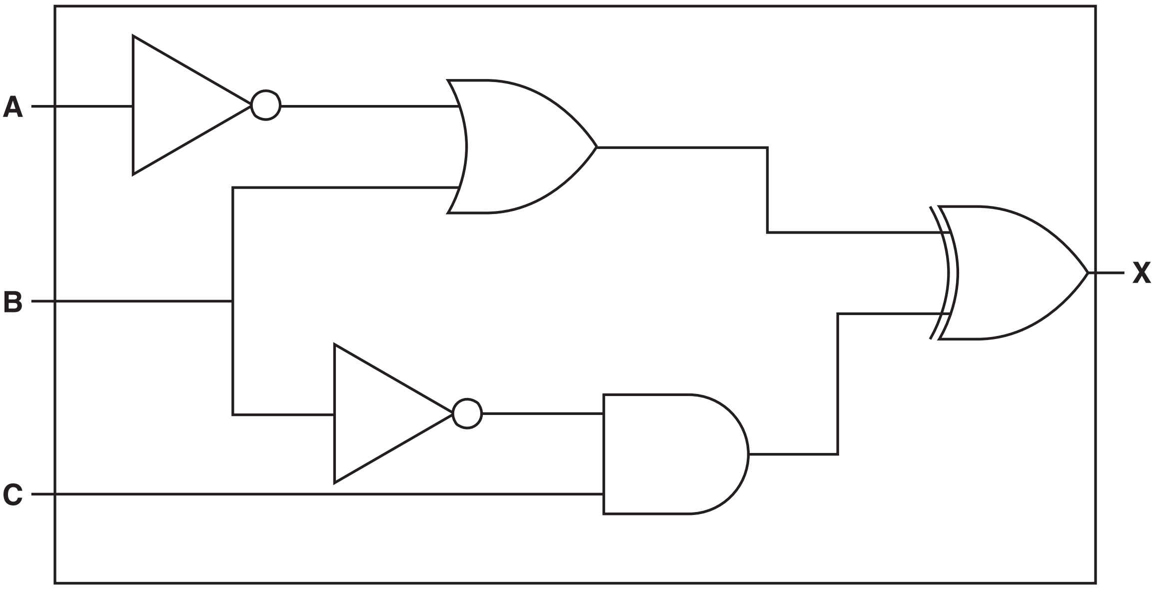
Off-line storage

Example ....

Use ....

....
[6]

7 **(a)** For this logic circuit:



Complete the truth table.

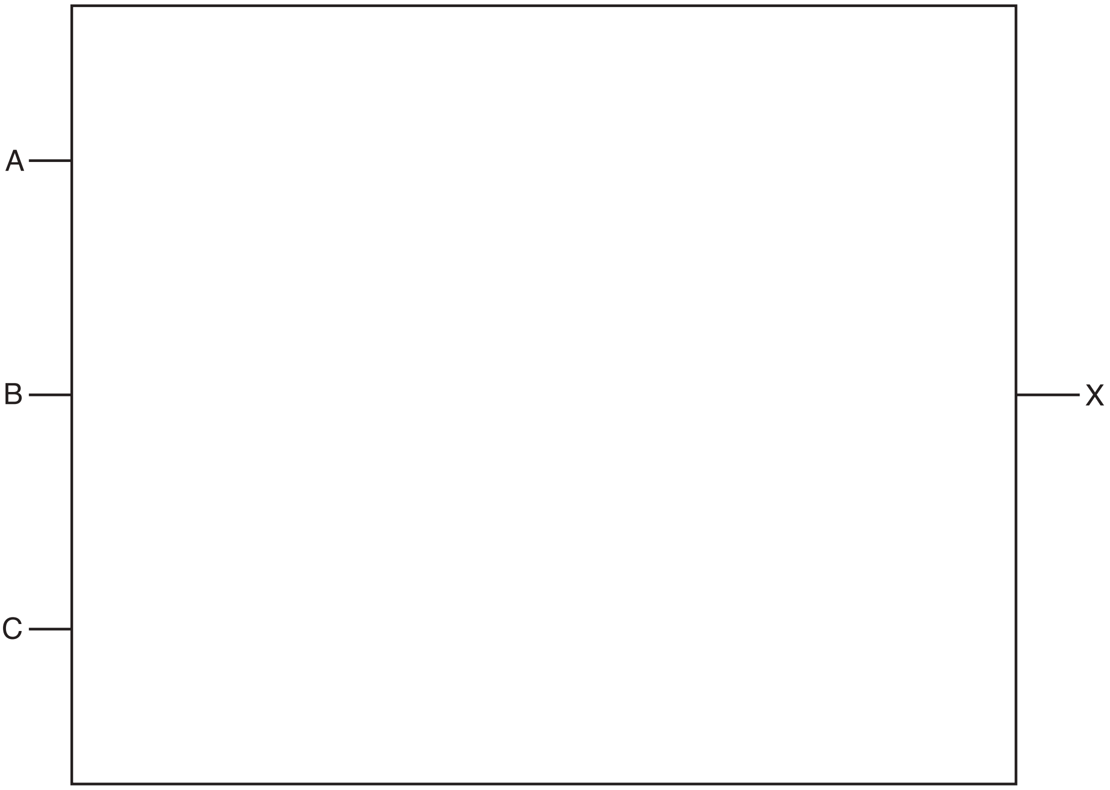
A	B	C	Working space	X
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

[4]

(b) For this logic statement:

X = 1 if (B is 1 OR C is NOT 1) AND ((A is NOT 1) AND (B is 1 OR C is 1))

Draw a logic circuit.



[6]

(c) Complete the truth table for the logic statement given in **part (b)**.

A	**B**	**C**	**Working space**	**X**
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

[4]

8 **(a)** **Three** descriptions and **two** methods of data transmission are given.

Tick (✓) the correct box to show the **Method** of data transmission for each description.

Description	**Method**	
	Serial	Parallel
Multiple bits are sent and received at the same time.		
Bits are sent one at a time in a single direction.		
Bits are sent using a single wire. Data can be sent or received, but not at the same time.		

[3]

(b) **Three** descriptions and **three** types of data transmission are given.

Tick (✓) the correct box to show the **Type** of data transmission for each description.

Description	**Type**		
	Simplex	Half-duplex	Duplex
Multiple bits are sent and received at the same time.			
Bits are sent one at a time in a single direction.			
Bits are sent using a single wire. Data can be sent or received, but not at the same time.			

[3]

9 A 32-second sound clip will be recorded. The sound will be sampled 16000 times a second.

Each sample will be stored using 8 bits.

Calculate the file size in kilobytes. **You must show all of your working.**

File Size .... kB

[3]

10 The table shows a segment of primary memory from a Von Neumann model computer.

Address	**Contents**
10001	11001101
10010	11110001
10011	10101111
10100	10000110
10101	00011001
10110	10101100

The program counter contains the data 10010.

(a) **(i)** State the data that will be placed in the memory address register (MAR).

....[1]

(ii) State the data that will be placed in the memory data register (MDR).

....[1]

(b) Describe the stored program concept when applied to the Von Neumann model.

....

....

....

....

....

....

....

....[4]

11 Miriam needs to use a large high-resolution photo as a thumbnail image on a website.

She will use lossy compression to reduce the file size of the photo to create the thumbnail image.

(a) State why a smaller file size is appropriate for this situation.

....

....[1]

(b) Explain how lossy compression reduces the file size.

....

....

....

....

....

....

....

....[4]

12 A hospital stores the results of medical tests on a computer system. Each patient is given a wristband containing a unique barcode. The barcode is used every time the patient has a medical test.

(a) Explain **two** benefits of using barcodes in this situation.

Benefit 1 ....

....

....

....

Benefit 2 ....

....

....

....
[4]

(b) Describe how the barcode is read.

....

....

....

....

....

....

....

....[4]

13 State **four** functions of an operating system.

Function 1 ....

....

Function 2 ....

....

Function 3 ....

....

Function 4 ....

....

[4]

Cambridge Assessment International Education
Cambridge International General Certificate of Secondary Education

COMPUTER SCIENCE **0478/22**

Paper 2 **March 2018**

MARK SCHEME

Maximum Mark: 50

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 March 2018 series for most Cambridge IGCSE$^{®}$, Cambridge International A and AS Level components and some Cambridge O Level components.

This syllabus is approved for use in England, Wales and Northern Ireland as a Cambridge International Level 1/Level 2 Certificate.

This document consists of **9** printed pages.

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
	Section A	
1(a)(i)	1 mark for name and 1 mark for appropriate data structure from **Task 1** e.g. name `studentName` Data structure Array	**2**
1(a)(ii)	1 mark for name, 1 mark for appropriate value and 1 mark for appropriate use. From any task e.g. Constant name `classSize` Value 30 Use Fixed number of students in a class Variable name `counter` Data type integer Use to count the number of student names entered	**6**
1(b)	1 mark for changing the value of a variable/constant/ counter, 1 mark for change to the program e.g. Value: Change the value of the variable/constant `maxGroupSize` in **Task 2** to 25 Program change: Oversubscribed test will be changed to test for over 50 // second group starts at 26	**2**

Question	Answer	Marks
1(c)	Any **five** from Initialise subject counts Loop through students Input and store student name… … and input and store two subject choices for that student Update subject counts // update subject lists of student names Error message if subject not found Output total subject counts Prompts for all inputs and messages with all outputs	**5**

Sample answer for **Task 1**

```
physics ← 0; chemistry ← 0; history ← 0; geography ← 0; compSci ← 0;
FOR counter ← 1 TO 60
   INPUT name, subject(1), subject(2)
   studentName(counter) ← name
   studentCounter1(counter) ← subject(1)
   studentCounter2(counter) ← subject(2)
   FOR sCount ← 1 TO 2
      CASE subject(sCount) OF
            'Physics': physics ← physics + 1
            'Chemistry': chemistry ← chemistry + 1
            'History': history ← history + 1
            'Geography': geography ← geography + 1
            'Computer Science': compSci ← compSci + 1
      OTHERWISE: PRINT 'Error subject not found'
      ENDCASE
   NEXT
NEXT
PRINT 'Group            Number of Students'
PRINT 'Physics          ',physics
PRINT 'Chemistry        ',chemistry
PRINT 'History          ',history
PRINT 'Geography        ',geography
PRINT 'Computer Science ',compSci
```

Question	Answer	Marks
1(d)	Any **five** from Explanation for calculation of total number of spare places Check the number of students who have chosen each subject If less than 10 no places available (Otherwise)If less than 20… …Calculate number of spare places (20 – number) If more than 20 and less than 40… …Calculate number of spare places (40 – number) Keep a running total of the total number of spare places	**5**

Question	Answer	Marks
Section B		

2(a)	1 mark for each error identified + suggested correction
	`NUMBERS` should be `Number`
	`IF Number > 100` should be `IF Number >= 100`
	`INPUT Number` is missing from inside the loop insert `INPUT Number` after the `IF` statement
	The final `PRINT Number` is not needed remove it
		4

2(b)	One mark for both ends of the range and correct inequality symbols and one mark for the **AND.** The test should be `IF Number >= 100 AND Number <= 200`	**2**

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>3</td><td>
<table>
<tr><th>Weight</th><th>totalWeight</th><th>totalNumber</th><th>OUTPUT</th></tr>
<tr><td></td><td>0</td><td>0</td><td></td></tr>
<tr><td>50</td><td>50</td><td>1</td><td></td></tr>
<tr><td>70</td><td>120</td><td>2</td><td></td></tr>
<tr><td>65</td><td>185</td><td>3</td><td></td></tr>
<tr><td>100</td><td>285</td><td>4</td><td></td></tr>
<tr><td>95</td><td>380</td><td>5</td><td></td></tr>
<tr><td>50</td><td>430</td><td>6</td><td></td></tr>
<tr><td>55</td><td>485</td><td>7</td><td></td></tr>
<tr><td>85</td><td>570</td><td>8</td><td></td></tr>
<tr><td>70</td><td>640</td><td>9</td><td>Lift overload, step out</td></tr>
</table>
One mark for each correct column.</td><td>4</td></tr>
</table>

Question	Answer	Marks
4	1 mark for value and 1 mark for appropriate reason e.g. Value 1 `2` (1) boundary should be accepted as weight OK (1) Value 2 `two` (1) erroneous/abnormal should be rejected (1)	**4**

Question	Answer	Marks

5	2 marks for appropriate explanation, 1 mark example programming statements showing sequence, 1 mark example programming statement(s) showing selection e.g.
	
	Sequence is the concept of one statement being executed after another(1) whereas selection decides which statement(s) are to be executed depending upon the result of a question (1)
	sequence example (1)
	`PRINT X`
	`PRINT Y`
	Selection example (1)
	`IF X > Y THEN PRINT X ELSE PRINT Y`
		4

Question	Answer	Marks
6(a)	1 mark for appropriate field name and appropriate data type, then 1 mark for appropriate explanation. e.g. Metal, type text(1) a single character/word that can be input accurately/quickly(1) Item, type text (1) a single character/word that can be input accurately/quickly(1) Number in Stock, type number (1) can be used for calculations (1) Price, type currency (1) properly formatted and can be used for calculations (1)	**8**
6(b)	All fields could contain duplicate values	**1**

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>6(c)</td><td>
<table>
<tr><td>Field:</td><td>Metal</td><td>Item</td><td>Number in Stock</td><td>Price</td></tr>
<tr><td>Table:</td><td>JEWEL</td><td>JEWEL</td><td>JEWEL</td><td>JEWEL</td></tr>
<tr><td>Sort:</td><td></td><td></td><td></td><td></td></tr>
<tr><td>Show:</td><td>☐</td><td>☐</td><td>☑</td><td>☑</td></tr>
<tr><td>Criteria:</td><td>=”silver”</td><td>=”bracelet”</td><td></td><td></td></tr>
<tr><td>or:</td><td></td><td></td><td></td><td></td></tr>
</table>
One mark for columns 1 + 2, one mark for columns 3 + 4
One mark for accuracy of syntax and spelling</td><td>3</td></tr>
</table>

Cambridge International Examinations
Cambridge International General Certificate of Secondary Education

---

COMPUTER SCIENCE **0478/22**

Paper 2 Problem-solving and Programming **February/March 2018**

PRE-RELEASE MATERIAL

No Additional Materials are required.

This material should be given to the relevant teachers and candidates as soon as it has been received at the Centre.

READ THESE INSTRUCTIONS FIRST

Candidates should use this material in preparation for the examination. Candidates should attempt the practical programming tasks using their chosen high-level, procedural programming language.

This document consists of **2** printed pages.

CAMBRIDGE
International Examinations

[Turn over

In preparation for the examination candidates should attempt the following practical tasks by **writing and testing a program or programs**.

Students in a school are allowed to choose extra subjects each year. Students provide the school administrator with their names and their subject choices. Places in subject groups are allocated on a 'first come, first served' basis. There are two classes of 30 students and they can each choose **two** extra subjects from:

- Physics
- Chemistry
- History
- Geography
- Computer Science

The maximum group size for each subject choice is 20 students and the minimum group size is 10 students. If more than 20 students choose a subject then that subject can be split into two groups. Each subject can have no more than two groups. If less than 10 students choose a subject then it is not available that year. A program is required to show a summary of the number of students who have chosen each subject, identify subject group sizes, produce subject group lists and identify problems.

Write and test a program or programs for the school administrator.

- Your program or programs must include appropriate prompts for the entry of data.
- Error messages and other output need to be set out clearly and understandably.
- All variables, constants and other identifiers must have meaningful names.

You will need to complete these **three** tasks. Each task must be fully tested.

TASK 1 – Data entry and number of students who have chosen each subject.

The school administrator enters the data for each student. Write a program for TASK 1 to store this data then calculate and output the number of students who have chosen each subject.

TASK 2 – Output subject group lists and identify problems.

Using your results from TASK 1, allocate students to subject groups. Print out list(s) of student names for each viable subject group. Identify any subjects that are over or undersubscribed, identify the students who have been allocated to one subject group only and those who have not been allocated to any group. Print out this information.

TASK 3 – Identify spare places in subject groups.

Using your results from TASK 2, print out the number of spare places for each subject. Any group that has fewer than 20 students has spare places. Calculate the total number of spare places and the total number of unallocated student choices. Show whether the number of spare places available is enough to cover the unallocated choices.

Cambridge International Examinations
Cambridge International General Certificate of Secondary Education

CANDIDATE NAME

CENTRE NUMBER			CANDIDATE NUMBER	

COMPUTER SCIENCE **0478/22**

Paper 2 Problem-solving and Programming **February/March 2018**

1 hour 45 minutes

Candidates answer on the Question Paper.

No Additional Materials are required.

No calculators allowed.

READ THESE INSTRUCTIONS FIRST

Write your Centre number, candidate number and name in the spaces at the top of this page.
Write in dark blue or black pen.
You may use an HB pencil for any diagrams, graphs or rough working.
Do not use staples, paper clips, glue or correction fluid.
DO **NOT** WRITE IN ANY BARCODES.

Answer **all** questions.

DO NOT ATTEMPT TASKS 1, 2 AND 3 in the pre-release material; these are for information only.

You are advised to spend no more than **40 minutes** on **Section A** (Question 1).

No marks will be awarded for using brand names of software packages or hardware.

At the end of the examination, fasten all your work securely together.

The number of marks is given in brackets [ ] at the end of each question or part question.

The maximum number of marks is 50.

This document consists of **11** printed pages and **1** blank page.

DC (LK/CGW) 148974/2

Section A

You are advised to spend no longer than 40 minutes answering this section.

Here is a copy of the pre-release material.

DO NOT attempt Tasks 1, 2 and 3 now.

Use the pre-release material and your experience from attempting the tasks before the examination to answer Question 1.

Pre-release material

Students in a school are allowed to choose extra subjects each year. Students provide the school administrator with their names and their subject choices. Places in subject groups are allocated on a ‘first come, first served’ basis. There are two classes of 30 students and they can each choose **two** extra subjects from:

- Physics
- Chemistry
- History
- Geography
- Computer Science

The maximum group size for each subject choice is 20 students and the minimum group size is 10 students. If more than 20 students choose a subject then that subject can be split into two groups. Each subject can have no more than two groups. If less than 10 students choose a subject then it is not available that year. A program is required to show a summary of the number of students who have chosen each subject, identify subject group sizes, produce subject group lists and identify problems.

Write and test a program or programs for the school administrator.

- Your program or programs must include appropriate prompts for the entry of data.
- Error messages and other output need to be set out clearly and understandably.
- All variables, constants and other identifiers must have meaningful names.

You will need to complete these **three** tasks. Each task must be fully tested.

TASK 1 – Data entry and number of students who have chosen each subject.

The school administrator enters the data for each student. Write a program for TASK 1 to store this data then calculate and output the number of students who have chosen each subject.

TASK 2 – Output subject group lists and identify problems.

Using your results from TASK 1, allocate students to subject groups. Print out list(s) of student names for each viable subject group. Identify any subjects that are over or undersubscribed, identify the students who have been allocated to one subject group only and those who have not been allocated to any group. Print out this information.

TASK 3 – Identify spare places in subject groups.

Using your results from TASK 2, print out the number of spare places for each subject. Any group that has fewer than 20 students has spare places. Calculate the total number of spare places and the total number of unallocated student choices. Show whether the number of spare places available is enough to cover the unallocated choices.

1 **(a)** All identifiers should have meaningful names.

(i) State the name and data structure that you have used to record student names in **Task 1**.

Name ....

Data structure ....
[2]

(ii) State the name of **one** constant and the name of **one** variable that you could have used in your programmed solution.
State the value that would be assigned to the constant. State the data type for the variable. Explain what each one would be used for in your programmed solution.

Constant name ....

Value ....

Use ....

....

Variable name ....

Data type ....

Use ....

....
[6]

(b) Explain how you would change your program for **Task 2** if the maximum group size for each subject is increased to 25.

....

....

....

....

....

....[2]

(c) Write an algorithm to complete **Task 1**, using **either** pseudocode, programming statements **or** a flowchart.

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....[5]

(d) Explain how your program calculates the total number of spare places in **Task 3**. Any programming statements shown in your answer must be fully explained.

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....[5]

Section B

2 An algorithm has been written in pseudocode to input some numbers and print out any numbers that are greater than or equal to `100`. The number `999` stops the algorithm.

```
INPUT Number
WHILE NUMBERS <> 999 DO
  IF Number > 100 THEN PRINT Number ENDIF
ENDWHILE
PRINT Number
```

(a) Find the **four** errors in the pseudocode and suggest corrections.

Error 1 ....

Correction ....

....

Error 2 ....

Correction ....

....

Error 3 ....

Correction ....

....

Error 4 ....

Correction ....

.... [4]

(b) Show, using pseudocode, how you would change the corrected algorithm to print out any numbers between `100` and `200` inclusive.

....

....

....

....

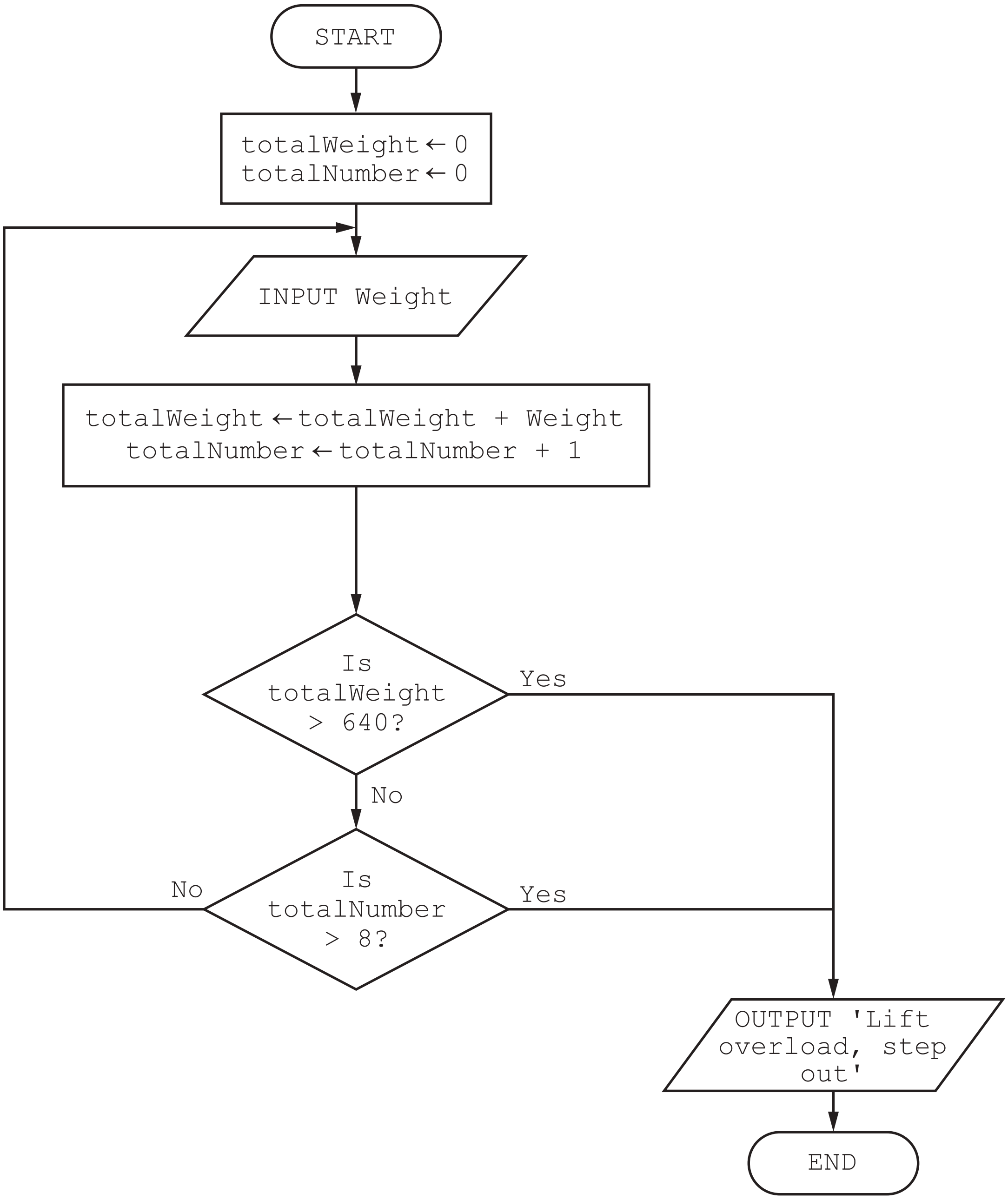
....

....

....

....[2]

3 This flowchart inputs the weight in kilograms of a passenger stepping into a lift. The lift can take a maximum of eight passengers or a maximum weight of 640 kilograms.



Complete the trace table for the passenger input data:

50, 70, 65, 100, 95, 50, 55, 85, 70, 75

Weight	totalWeight	totalNumber	OUTPUT

[4]

4 A program checks if the weight of a baby is at least 2 kilograms.

Give, with reasons, **two** different values of test data that could be used for the baby's weight. Each reason must be different.

Value 1 ....

Reason ....

....

Value 2 ....

Reason ....

....
[4]

5 Explain the difference between the programming concepts of **sequence** and **selection**. Include an example of a programming statement for each concept in your explanation.

....

....

....

....

....

....

....

....

....[4]

6 A database table, JEWEL, is used to keep a record of jewellery for sale in a shop. Each item of jewellery can be made of silver, platinum or gold metal. The shop stocks rings, bracelets and necklaces. The number in stock and the price is also stored.

(a) Identify the **four** fields required for the database. Give each field a suitable name and data type. Explain why you chose the data type for each field.

Field 1 Name .... Data type ....

Explanation ....

....

Field 2 Name .... Data type ....

Explanation ....

....

Field 3 Name .... Data type ....

Explanation ....

....

Field 4 Name .... Data type ....

Explanation ....

....
[8]

(b) Explain why none of these fields could be used as a primary key.

....

....[1]

(c) Using the query-by-example grid below, write a query to identify the silver bracelets. Only display the number in stock and the price.

Field:				
Table:				
Sort:				
Show:	☐	☐	☐	☐
Criteria:				
or:				

[3]

BLANK PAGE

Cambridge Assessment
International Education

Grade thresholds – June 2018

Cambridge IGCSE™ Computer Science (0478)

Grade thresholds taken for Syllabus 0478 (Computer Science) in the June 2018 examination.

		minimum raw mark required for grade:						
	maximum raw mark available	A	B	C	D	E	F	G
Component 11	75	47	36	26	21	15	10	5
Component 12	75	45	35	26	21	15	10	5
Component 13	75	45	35	26	20	15	10	5
Component 21	50	31	23	15	12	10	7	4
Component 22	50	29	21	14	11	9	6	3
Component 23	50	32	25	18	15	12	9	6

Grade A* does not exist at the level of an individual component.

The maximum total mark for this syllabus, after weighting has been applied, is **125**.

The overall thresholds for the different grades were set as follows.

Option	Combination of Components	A*	A	B	C	D	E	F	G
AX	11, 21	95	77	59	41	33	25	17	9
AY	12, 22	90	73	56	40	32	24	16	8
AZ	13, 23	92	76	60	44	35	27	19	11

Cambridge IGCSE

Cambridge Assessment International Education
Cambridge International General Certificate of Secondary Education

COMPUTER SCIENCE **0478/11**

Paper 1 **May/June 2018**

MARK SCHEME

Maximum Mark: 75

Published

This document consists of **12** printed pages.

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:

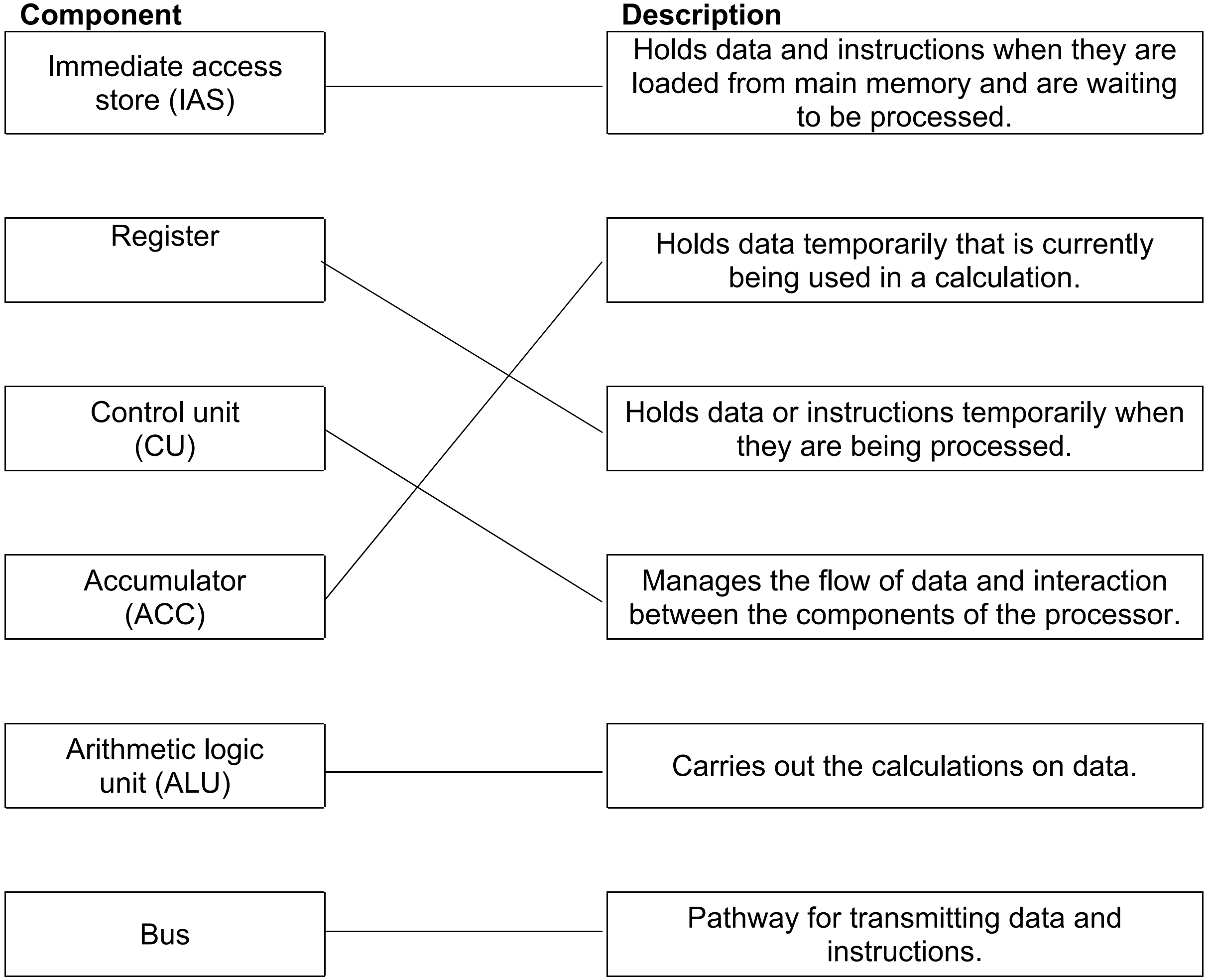
Marks awarded are based solely on the requirements as defined in the mark scheme. Marks should not be awarded with grade thresholds or grade descriptors in mind.

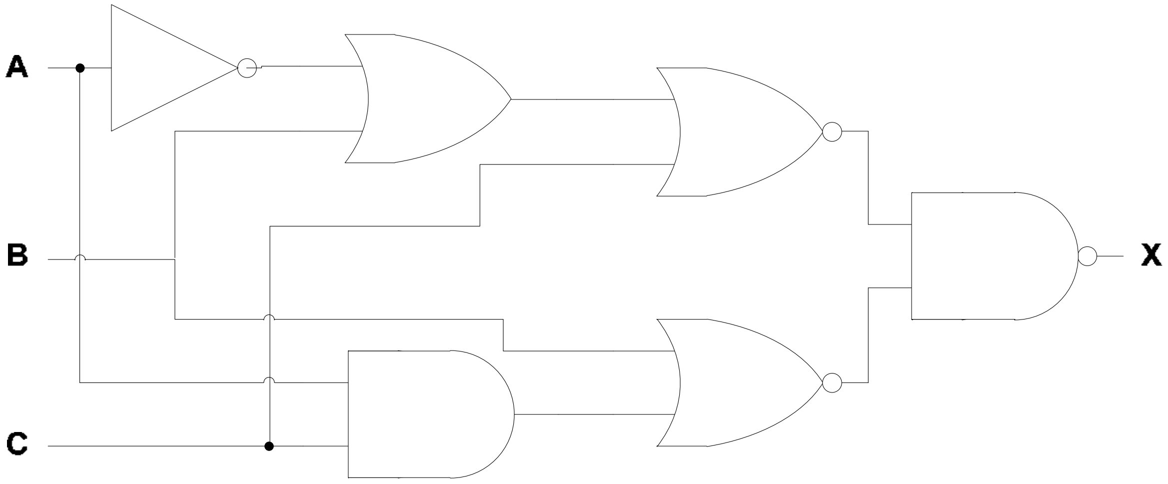
Question	Answer	Marks
1	1 mark for each correct answer, in the given order: – analogue – digital – denary – 10 – binary – 2	**6**

Question	Answer	Marks
2	1 mark for each correct conversion: – 42 – 257 – 542	**3**

Question	Answer	Marks
3	1 mark for correct register, 3 marks for reason: – Register C Any **three** from: – Count the number of 1/0 bits (in each byte/register) – Two bytes/registers have an odd number of 1/0 bits // Two use odd parity – Odd parity must be the parity used – One byte/register has an even number of 1/0 bits // One uses even parity – One with an even number of one bits/even parity is incorrect // Register C should have odd parity	**4**

Question	Answer	Marks
4(a)	1 mark for each correct answer: Lossy (compression) Lossless (compression)	**2**
4(b)	1 mark for correct compression, 3 marks for description: – Lossless (compression) Any **three** from: – The file can be restored/decompressed to the exact same state it was before compression/ to original – (It is a computer program so) no data can be lost // Lossy would remove data – Will not run correctly (with any other compression) – (Lossless) will give repeating words/sections of word a value// RLE is used // Other valid examples of methods of lossless compression – Value is recorded in an index	**4**

Question	Answer	Marks
5	1 mark for each correct line, up to a maximum of 5 marks: 	**5**

Question	Answer	Marks
6(a)	1 mark for each correct logic gate (with the correct direction of input(s)) 	**6**

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>6(b)</td><td>4 marks for 8 correct outputs
3 marks for 6 or 7 correct outputs
2 marks for 4 or 5 correct outputs
1 mark for 2 or 3 correct outputs
<table>
<tr><th>A</th><th>B</th><th>C</th><th>Working space</th><th>X</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td></td><td>1</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>1</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td></td><td>1</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>
</table>

Question	Answer	Marks
7	Compiler Any **three** from: – Translates high-level language into **machine code/low level language** – Translates (the source code) all in one go/all at once – Produces an executable file – Produces an error report Interpreter Any **three** from: – Translates high-level language into **machine code/low level language** – Translates (the source code) line by line/statement by statement – Stops if it finds an error – Will only continue when error is fixed	**6**

Question	Answer	Marks
8(a)	Any **four** from: – Shines light / (red) laser at barcode – Light is called an illuminator – Light is reflected back // White lines reflect light // Black lines reflect less light/absorbs light – Sensors / photoelectric cells detect the light – Different reflections / bars will give different binary values / digital values // pattern converted to digital values – A microprocessor interprets the data	**4**
8(b)	Any **three** from: – barcode identifies a (unique) product – barcode can be used to look up product (in a database) – data about stock levels can be stored on a system – stock can be automatically deducted from the system – can check stock is below a certain level // check stock level – automatic re-order // Alerts when stock is low – automatically update new stock level – to locate if an item of stock is available in another location	**3**

Question	Answer	Marks
8(c)	Any **four** from: – (Infrared) rays are sent across screen (from the edges) – Has sensors around edge // Sensors capture beams – (Infrared) rays form a grid across the screen – (Infrared) ray is broken (by a finger blocking a beam) – **Calculation** is made (on where beam is broken) to locate the 'touch' // Co-ordinates are used to locate the touch	**4**
8(d)	Secondary Storage – any **two** from: – Not directly accessed by the CPU – Non-volatile storage – Secondary is internal to the computer/device – An example of secondary storage would be HDD/SSD Off-line storage – any **two** from: – Non-volatile storage – Off-line storage is storage that is removable from a computer/device // not internal // portable – An example of off-line storage would be CD/DVD/USB stick/SD card/magnetic tape/ external HDD/SSD	**4**

Question	Answer	Marks
9	Any **six** from: – Suitable biometric device, such as fingerprint scanner/retina/eye/iris scanner/face recognition/voice recognition/palm scanner // description of use e.g. use fingerprint on device – Sensor (in biometric device) captures/takes data/readings (of user) – Data/readings are converted from analogue to **digital** (using ADC) – Data/reading sent to the microprocessor – Data/readings compared to stored values/data … – … if data/readings **match** user can enter – … if data/readings **do not match** user is declined entry // user asked to try again … – … alert may be sent to security // alarm may sound	**6**

Question	Answer	Marks
10(a)	Any **four** from: – Structure **and** presentation are defined using (mark-up) tags – Structure **and** presentation dictate the appearance of the website – Structure is used for layout – Example of structure – Presentation is used for formatting / style – Example of formatting – Separate file / CSS can be used for presentation content	**4**
10(b)(i)	1 mark for each correct part – domain (name) – file name/webpage name	**2**
10(b)(ii)	Any **two** from: – Hypertext Transfer Protocol Secure // it is the access protocol // It is a protocol – It means the website uses SSL/TLS – It means data sent (to and from the webserver) is encrypted	**2**
10(c)	Any **two** from e.g. : – To store items that a customer has added to an online shopping basket – To store a customer's credit card details – To store log-in details – To track what product a customer browses // Track music preferences – Targeted advertising // making recommendations – Personalises/customises the experience – Shows who are new and returning customers – To speed up log-in times – To speed up/allow single click purchases – Improves the experience	**2**

Question	Answer	Marks
10(d)	Any **four** from: – Prevents direct access to the **webserver** // Sits between **user** and **webserver** – If an attack is launched it hits the proxy server instead // can be used to help prevent DDOS // help prevent hacking of **webserver** – Used to direct invalid traffic away from the webserver – Traffic is examined by the proxy server // Filters traffic – If traffic is valid the data from the webserver will be obtained by the user – If traffic is invalid the request to obtain data is declined – Can block requests from certain IP addresses	**4**

Cambridge Assessment International Education
Cambridge International General Certificate of Secondary Education

COMPUTER SCIENCE **0478/12**

Paper 1 **May/June 2018**

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 May/June 2018 series for most Cambridge IGCSE™, Cambridge International A and AS Level and Cambridge Pre-U components, and some Cambridge O Level components.

This syllabus is approved for use in England, Wales and Northern Ireland as a Cambridge International Level 1/Level 2 Certificate.

This document consists of **10** printed pages.

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	1 mark for each unit, in the given order: – nibble – byte – megabyte (MB) – gigabyte (GB)	**4**

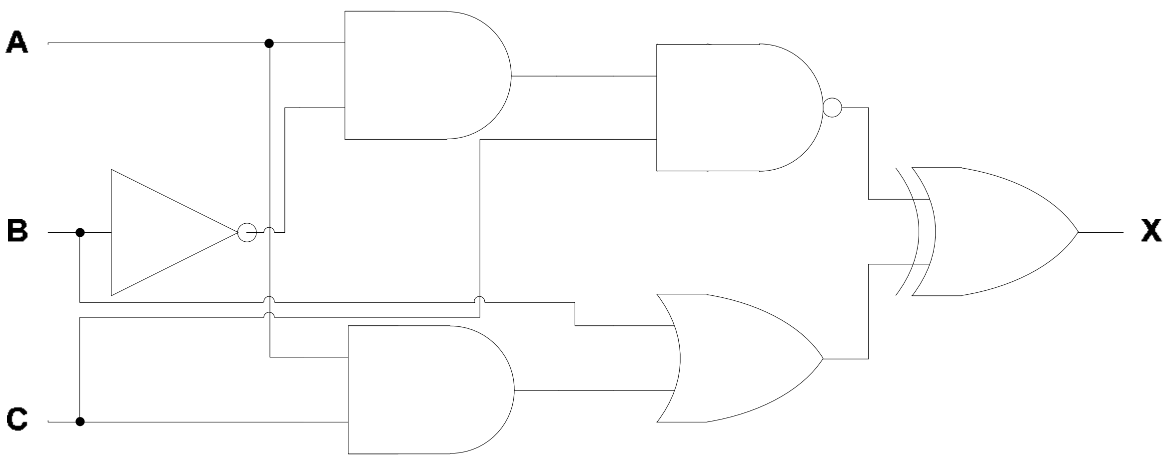
Question	Answer	Marks
2(a)	Any **four** from: – Image is converted from <u>analogue</u> to digital (using ADC) – Image is turned into pixels – Each pixel is given a binary value – Pixels form a grid (to create the image) – Each pixel has a colour – Pixels are stored in sequence (in a file) – Meta data is stored (to describe the dimensions/resolution of the image) // It stores the dimensions/colour depth .etc. – An example of a suitable photo file format e.g. JPEG	**4**
2(b)	1 mark for correct compression, 3 marks for explanation: – Lossy Any **three** from: – Lossy would reduce the file size **more** (than lossless) – The **redundant** data can be removed from the files // by example (must be about redundant data) – Images can still be a similar quality – There is no requirement for the files to be exactly the same as original file – Photos can be sent quick**er** // fast**er** to upload // fast**er** to download	**4**

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>3(a)</td><td>1 mark for each correct register
<table>
<tr><td>Hours</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td>Minutes</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>Seconds</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td></tr>
</table></td><td>3</td></tr>
<tr><td>3(b)</td><td>1 mark for each correct section:
<table>
<tr><td>0</td><td>5</td><td>:</td><td>2</td><td>6</td><td>:</td><td>5</td><td>5</td></tr>
<tr><td colspan="2">Hours</td><td></td><td colspan="2">Minutes</td><td></td><td colspan="2">Seconds</td></tr>
</table></td><td>3</td></tr>
</table>

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>4</td><td>1 mark for each correct section:
<table>
<tr><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td colspan="4">⟵ 1 mark ⟶</td><td colspan="4">⟵ 1 mark ⟶</td><td colspan="4">⟵ 1 mark ⟶</td></tr>
</table></td><td>3</td></tr>
</table>

Question	Answer	Marks
5	1 mark for correct register, 3 marks for reason: – Register Y Any **three** from: – Count the number of 1/0 bits (in each byte/register) – Two bytes/registers have an odd number of 1/0 bits // Two have odd parity – Even parity must be the parity used – One byte/register has an even number of 1/0 bits // One uses even parity – The two with an odd number of one bits/odd parity are incorrect // Register X and Z should have even parity	**4**

Question	Answer	Marks
6	1 mark for each correct missing word, in the given order: – fetches – immediate access store // IAS – program counter // PC – memory address register // MAR – memory data register // MDR – executed – arithmetic logic unit // ALU – accumulator // ACC	**8**

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>7(a)</td><td>1 mark for each correct logic gate with correct direct of input(s):
 </td><td>6</td></tr>
<tr><td>7(b)</td><td>4 marks for 8 correct outputs
3 marks for 6 or 7 correct outputs
2 marks for 4 or 5 correct outputs
1 mark for 2 or 3 correct outputs
<table>
<tr><th>A</th><th>B</th><th>C</th><th>Working space</th><th>X</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td></td><td>1</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>1</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>0</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td></td><td>0</td></tr>
</table></td><td>4</td></tr>
</table>

Question	Answer	Marks
8	1 mark for correct translator, 3 marks for explanation: – Compiler Any **three** from: – Does not require recompilation // compiled program can be executed without a compiler … – ... therefore, allows faster execution – Provides an executable file … – … therefore, allows him to just send machine code – **Dimitri's friend** does not need translation/compilation **software** to execute the program	**4**

Question	Answer	Marks
9(a)	QR/Quick response	**1**
9(b)	Any **four** from: – Read/scanned using **app** (on mobile device) – It is the **camera** that is used to scan/capture the image – The three large squares are used to define the alignment // uses alignment targets/modules – Black squares reflect less light // white squares reflect more light – The app/device processes the image – Each small square/pixel is converted to a binary value	**4**

Question	Answer	Marks
10(a)	Any **four** from: – Conductive layer – An electrostatic/electric field is created – **Sensor(s)** (around the screen) monitor the electrostatic field – When touched (electrostatic) charge is transferred to finger – Location of touch is **calculated** // Co-ordinates used to **calculate** touch	**4**

Question	Answer	Marks
10(b)(i)	Any **two** from: – Gloves are not conductive // Gloves are an insulator – Block current/charge from finger / body / person – Stop the electrostatic field being disturbed/changed	**2**
10(b)(ii)	Any **two** from e.g. (1 mark for method, 1 for expansion): – She could use a (conductive) stylus… – … this will allow the charge to be charged/disturbed – She could use capacitive gloves… – … this will allow the charge to be charged/disturbed – She could use a natural language interface/voice operated interface … – … she could give vocal commands to the device	**2**

Question	Answer	Marks
11	Any **six** from: – Suitable sensor (motion/infra-red) – Data converted (from analogue) to digital (using ADC) – Data sent to microprocessor – Data is compared to stored value/range … – … if data matches/out of range data security light turned on … – … waits for suitable period/until no motion detected … – … light turned off – Continuous loop/process	**6**

Question	Answer	Marks
12(a)(i)	Encryption	**1**

Question	Answer	Marks
12(a)(ii)	Any **five** from: – Her personal details before encryption is the plain text – The plain text/her personal details is encrypted using an encryption algorithm – The plain text/her personal details is encrypted using a key – The encrypted text is cypher/cipher text – The key is transmitted **separately** (from the text) – The key is used to decrypt the cypher text (after transmission)	**5**
12(b)	Any **three** from a single error method: – Checksum – Calculation carried out on data – (checksum/calculated) value sent with data – recalculated after transmission **and** compared to original – If they do not match an error is present – ARQ – uses acknowledgment **and** timeout – A request is sent with data to acknowledge all data is received – Acknowledgement sent back to say all data is received – If no acknowledgement is received in a time frame an error in transmission detected / data automatically resent.	**3**

Cambridge Assessment International Education
Cambridge International General Certificate of Secondary Education

---

COMPUTER SCIENCE **0478/13**

Paper 1 **May/June 2018**

MARK SCHEME

Maximum Mark: 75

Published

This syllabus is approved for use in England, Wales and Northern Ireland as a Cambridge International Level 1/Level 2 Certificate.

This document consists of **12** printed pages.

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**:

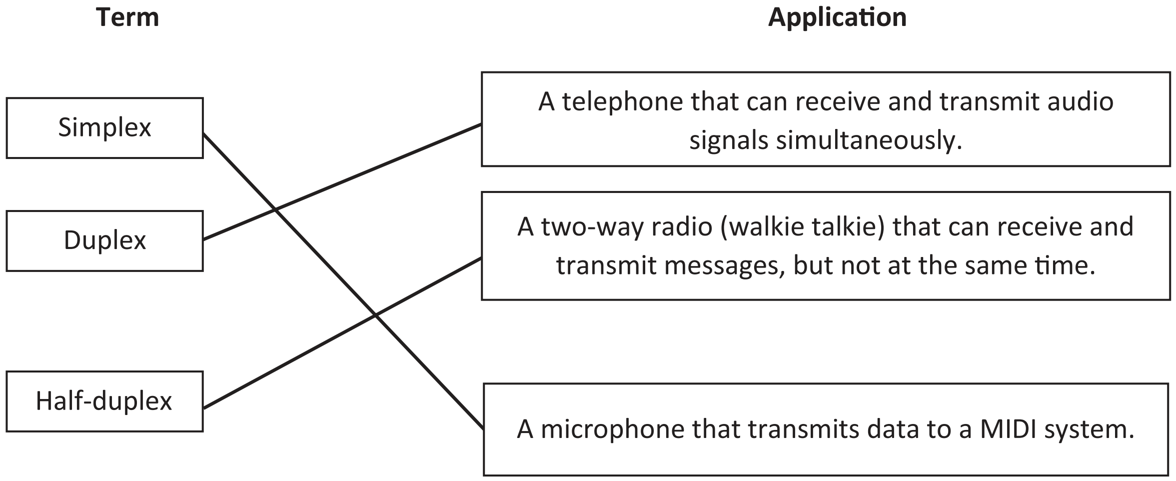
- marks are awarded for correct/valid answers, as defined in the mark scheme. However, credit is given for valid answers which go beyond the scope of the syllabus and mark scheme, referring to your Team Leader as appropriate
- marks are awarded when candidates clearly demonstrate what they know and can 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.

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>1</td><td>One mark per each sensor (sensors must be different):
<table>
<tr><th>Application</th><th>Sensor</th></tr>
<tr><td>Weighing a baby in a hospital</td><td>Pressure</td></tr>
<tr><td>Turning off a kettle when it boils</td><td>Temperature</td></tr>
<tr><td>Controlling an automatic door</td><td>Infrared / Light / Pressure</td></tr>
<tr><td>Monitoring the air quality in an aeroplane</td><td>Oxygen / Gas / Humidity</td></tr>
<tr><td>Counting cars crossing a bridge</td><td>Pressure / Infrared / Magnetic</td></tr>
</table>
</td><td>5</td></tr>
</table>

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>2</td><td>

Three correct lines = 2 marks
Two or one correct line = 1 mark</td><td>2</td></tr>
</table>

Question	Answer	Marks
3	2 marks per issue from: Phishing – Legitimate looking emails sent to use – When user clicks on attachment / link sent to fraudulent website – Asked to reveal/designed to steal sensitive information Pharming – Malicious code loaded on user hard drive – Will redirect URL requests to fraudulent website – Asked to reveal/designed to steal sensitive information Spam – Junk / unwanted email – Sent to large numbers of people – Used for advertising / spreading malware – Fills up mail boxes	**6**

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>4(a)(i)</td><td>
<table>
<tr><th>Received Byte</th><th>Transmitted correctly (✓)</th><th>Transmitted incorrectly (✓)</th></tr>
<tr><td>10001011</td><td></td><td>✓</td></tr>
<tr><td>10101110</td><td>✓</td><td></td></tr>
<tr><td>01011101</td><td>✓</td><td></td></tr>
<tr><td>00100101</td><td>✓</td><td></td></tr>
</table>
</td><td>4</td></tr>
<tr><td>4(a)(ii)</td><td>One from:
– ARQ
– Check Sum</td><td>1</td></tr>
</table>

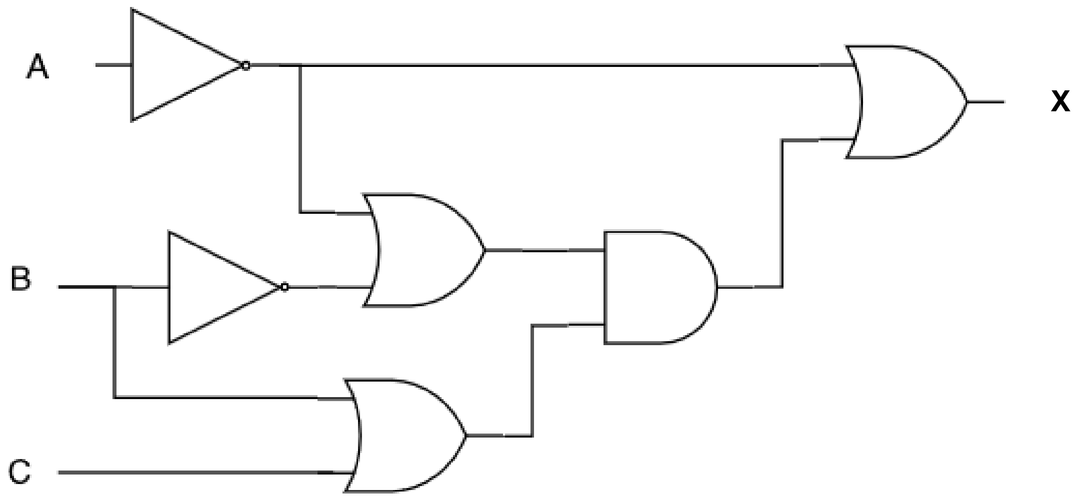
Question	Answer	Marks
4(b)(i)	– Multiple bits / byte(s) sent at the same time – Using multiple wires	**2**
4(b)(ii)	Any **one** from e.g.: – Integrated Circuits – Any appropriate CPU buses – Any suitable device connection that uses parallel	**1**
4(b)(iii)	Two from: – Bits remain synchronised … – … reducing data errors – Only single wire is required … – … more cost effective to install/manufacture	**2**
4(c)(i)	– Encrypted text is meaningless – Need the key to decrypt the text	**2**
4(c)(ii)	– Increase length / more bits used for key … – … will generate more possibilities for key / less chance of decryption by brute force method	**2**

Question	Answer	Marks
5(a)	(0)1101011	**1**
5(b)	000 100101100 1 mark for three leading zeros, 1 mark for correct binary number	**2**
5(c)	B3 1 mark for each correct character	**2**
6(a)	Any **two** from: – A signal sent from a device / software – Requests processor time // Processor stops to service interrupt – Interrupts have different priorities	**2**

Question	Answer	Marks
6(b)	Any **three** from e.g.: – Keyboard – Printer – Mouse	**3**

Question	Answer	Marks
7(a)	Any **three** from: – Does not require peripherals (mouse or keyboard) – Number of possible inputs limited / menu driven interface – Less chance of input error – Resistant to weather	**3**
7(b)	– Uses two/multiple layers – When top layer touched / pushed two layers make contact – Circuit is completed when layers touch – Point of contact is determined/calculated	**4**

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>8(a)</td><td>
<table>
<tr><td>A</td><td>B</td><td>C</td><td>X</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>1</td></tr>
</table>
All 8 for 4 marks
6 or 7 for 3 marks
4 or 5 for 2 marks
2 or 3 for 1 mark</td><td>4</td></tr>
</table>

Question	Answer	Marks
8(b)	 1 mark per gate with correct inputs	**6**

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>9</td><td>
<table>
<tr><th>Statement</th><th>Assembler
(✓)</th><th>Compiler
(✓)</th><th>Interpreter
(✓)</th></tr>
<tr><td>Translates high-level language into machine code</td><td></td><td>✓</td><td>✓</td></tr>
<tr><td>Provides error diagnostics</td><td>✓</td><td>✓</td><td>✓</td></tr>
<tr><td>Translates whole program to object code in one operation</td><td>✓</td><td>✓</td><td></td></tr>
<tr><td>Translates and executes one line of code at a time</td><td></td><td></td><td>✓</td></tr>
</table>
1 mark for each correct column</td><td>3</td></tr>
</table>

Question	Answer	Marks
10	Any **six** from: – Program counter (PC) holds address / location of the instruction – The address held in PC is sent to MAR – Address is sent using address bus – PC is incremented – The instruction is sent from address in memory to MDR – Instruction is transferred using the data bus – Instruction sent to CIR	**6**

Question	Answer	Marks
11	Any **three** from: – Optical media – Non-volatile – Offline – Single (continuous spiral) track – Data stored using lands / pits – Read using (red) lasers – Can be read only (R) or read write (RW)	**3**

Question	Answer	Marks
12	$256 \times 200 = \quad 51\,200$ $\frac{51\,200 \times 16}{8} = 102\,400$ $\frac{102\,400}{1024} = 100$ Answer 100 kB One mark for correct answer and two marks for correct calculations.	**3**

Question	Answer	Marks
13	Any **six** from e.g.: – Provide access to the internet / dial up / broadband – Usually charge a monthly fee – Monitor usage – Give users an IP address – Determine bandwidth – Supports domain names – Provide security services – Provide web hosting facilities – Provide access to Email / Mailbox – Provides online data storage	**6**

Cambridge Assessment International Education
Cambridge International General Certificate of Secondary Education

---

COMPUTER SCIENCE **0478/21**

Paper 2 **May/June 2018**

MARK SCHEME

Maximum Mark: 50

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 May/June 2018 series for most Cambridge IGCSE™, Cambridge International A and AS Level and Cambridge Pre-U components, and some Cambridge O Level components.

This syllabus is approved for use in England, Wales and Northern Ireland as a Cambridge International Level 1/Level 2 Certificate.

This document consists of **11** printed pages.

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
	Section A	
1(a)(i)	Many correct answers, they must be meaningful. The following is an example only: **One** mark per bullet point • Data structure Array • Name `processor` • Data type string • Use to store processors currently available	**4**
1(a)(ii)	**One** mark per bullet point • Data structure given (1) • Data type (1) • Sample data (1) • More than one data structure described (1) Many correct answers, they must be meaningful. The following is an example only: e.g. Three arrays containing string data with name, address and phone number – John Smith, Cambridge, 01223 123456	**4**
1(b)	**One** mark for method, **one** mark for an extension or reason. Many correct answers, an example is given. Use a previously stored number//generates/uses an initial value (1) Update it (by 1) every time an estimate is made (1)	**2**

Question	Answer	Marks
1(c)	Any **five** from: 1 Initialise (stock level) flag 2 Check stock level for the chosen processor type 3 Only check RAM if processor available // Only check processor if RAM available 4 Check stock level for the chosen type of RAM 5 Finish process if problem with (RAM/Processor) stock levels 6 Identify out of stock (processor/RAM)//Set flag to appropriate value 7 Identify stock level OK//Set flag to appropriate value	**5**

Question	Answer	Marks
1(c)	Sample answer:	

```
foundProc ← FALSE
count ← 1
WHILE NOT foundProc AND count <=3 DO
  IF processor(estNo) = proc(count) AND stProc(count) > 0
    THEN
      foundProc ← TRUE
  ENDIF
  count ← count + 1
ENDWHILE
IF foundProc
  THEN
    foundRAM ← FALSE
    IF RAM(estNO) = RAM1 AND stRAM1 >0
      THEN
        foundRAM ← TRUE
        stRAM1 ← stRAM1 - 1
    ENDIF
    IF RAM(estNO) = RAM2 AND stRAM2 >0
      THEN
        foundRAM ← TRUE
        stRAM2 ← stRAM2 - 1
    ENDIF
ENDIF
IF NOT foundProc
  THEN
    OUTPUT "Processor out of stock"
  ELSE
    stProc(count) ← stProc(count) - 1
ENDIF
IF NOT foundRAM
  THEN
    OUTPUT "RAM out of stock"
ENDIF
```

Question	Answer	Marks
1(d)	**One** mark for each correct point (max 5): Explanation 1 How the number of orders was calculated 2 Deal with the case where the estimate has not been turned into an order 3 Calculating the total number of each component sold 4 Details of method actually used to calculate numbers of components 5 How the total value of all the orders was calculated 6 Display summary 7 Display complete summary of number of orders, total number of components and total value of orders Programming statements can be used but **must be explained** to gain credit.	**5**

Question	Answer	Marks
Section B		
2(a)	Any **six** from: 1 Initialisation of counters for positive numbers and zeros 2 Appropriate loop for 1000 iterations 3 Input number inside loop 4 Test for positive numbers 5 Update positive number counter 6 Test for zeros 7 Update zero counter 8 Output counters with appropriate messages outside loop	6

```
zero ← 0
posCount ← 0
FOR count ← 1 TO 1000
  INPUT number
  IF number > 0
    THEN posCount ← posCount + 1
  ENDIF
  IF number = 0
    THEN zero ← zero + 1
  ENDIF
NEXT
OUTPUT posCount, " positive numbers"
OUTPUT zero, " zeros"
```

Question	Answer	Marks
2(b)	Reduce the number of iterations to a manageable amount // Simulate the input (e.g. random generation)	1

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>3(a)</td><td>
<table>
<tr><th>Digit(1)</th><th>Digit(2)</th><th>Digit(3)</th><th>Digit(4)</th><th>Digit(5)</th><th>Digit(6)</th><th>Digit(7)</th><th>Digit(8)</th><th>Sum</th><th>OUTPUT</th></tr>
<tr><td>5</td><td>7</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>6</td><td>44</td><td>GTIN-8</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>57012346</td></tr>
</table>
<table>
<tr><th>Digit(1)</th><th>Digit(2)</th><th>Digit(3)</th><th>Digit(4)</th><th>Digit(5)</th><th>Digit(6)</th><th>Digit(7)</th><th>Digit(8)</th><th>Sum</th><th>OUTPUT</th></tr>
<tr><td>4</td><td>3</td><td>1</td><td>0</td><td>2</td><td>3</td><td>1</td><td>0</td><td>30</td><td>GTIN-8</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>43102310</td></tr>
</table>
One mark for data entry – both sets of digits 1–7
One mark for both Digit(8)
One mark for each Sum (max Two)
One mark for both OUTPUT
</td><td>5</td></tr>
<tr><td>3(b)</td><td>
Any three from
1 Change first loop to 8 iterations
2 Check that the input <code>Digit(8)</code> is equal to the calculated <code>Digit(8)</code>…
3 … if equal output check digit correct
4 … otherwise output check digit incorrect

Or

1 Change first loop to 8 iterations
2 Put all 8 digits through the algorithm to calculate <code>Sum</code> …
3 … if <code>MOD(Sum,10)</code> is equal to zero, check digit correct
4 … otherwise output check digit incorrect
</td><td>3</td></tr>
</table>

Question	Answer	Marks
4	**One** mark for each (max **three**) *10.00* boundary/erroneous data // the price should be rejected // value is out of range *9.99* boundary/extreme/normal data // the prices should be accepted // value is within normal range *ten* erroneous/abnormal data // input should be rejected // value is wrong type	**3**

Question	Answer	Marks
5	There are many possible answers. e.g.: Totalling is used to sum a list of numbers (1) Counting is used to find how many numbers/items there are in a list. (1) Totalling example (1) e.g. `Total = Total + Number` Counting example (1) e.g. `Counter = Counter + 1`	**4**

Question	Answer	Marks
6(a)	Fields 5 Records 8	**2**
6(b)	Any **two** from: Length check Type check Presence check Format check	**2**

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>6(c)</td><td>
<table>
<tr><td>Field:</td><td>Type</td><td>Sold Out</td><td>Date</td><td>Title</td></tr>
<tr><td>Table:</td><td>PERFORMANCE</td><td>PERFORMANCE</td><td>PERFORMANCE</td><td>PERFORMANCE</td></tr>
<tr><td>Sort:</td><td></td><td></td><td></td><td></td></tr>
<tr><td>Show:</td><td>☐</td><td>☐</td><td>☑</td><td>☑</td></tr>
<tr><td>Criteria:</td><td>Like “Jazz”</td><td>False</td><td></td><td></td></tr>
<tr><td>or:</td><td></td><td></td><td></td><td></td></tr>
</table>

One mark per correct column.</td><td>**4**</td></tr>
</table>

Cambridge Assessment International Education
Cambridge International General Certificate of Secondary Education

---

COMPUTER SCIENCE **0478/22**

Paper 2 **May/June 2018**

MARK SCHEME

Maximum Mark: 50

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 May/June 2018 series for most Cambridge IGCSE™, Cambridge International A and AS Level and Cambridge Pre-U components, and some Cambridge O Level components.

This syllabus is approved for use in England, Wales and Northern Ireland as a Cambridge International Level 1/Level 2 Certificate.

This document consists of **7** printed pages.

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:

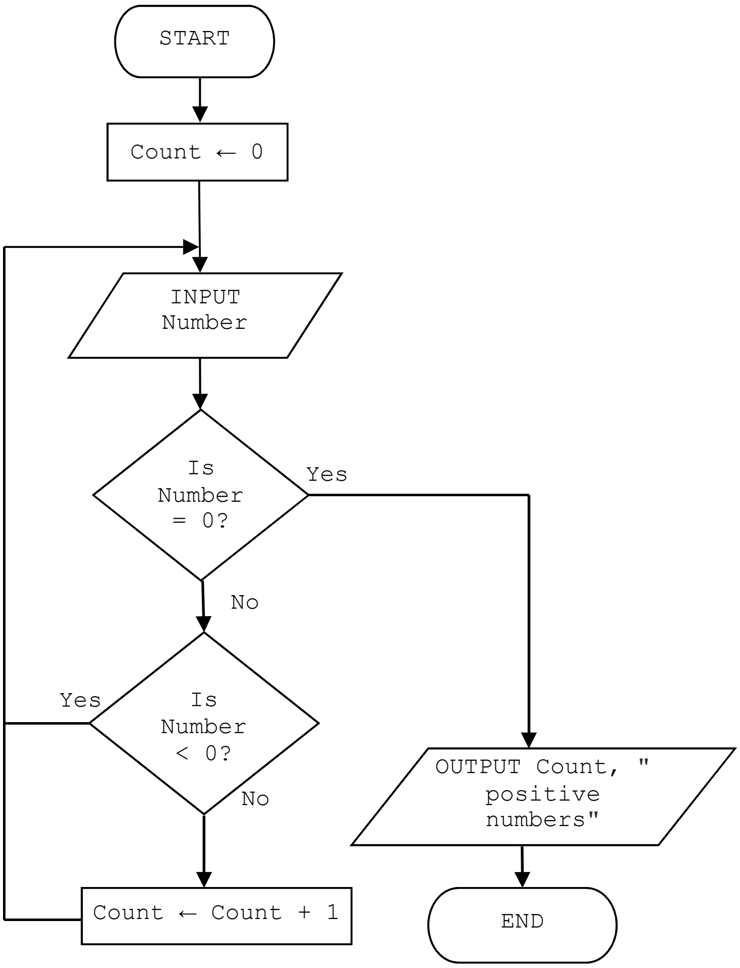
Marks awarded are based solely on the requirements as defined in the mark scheme. Marks should not be awarded with grade thresholds or grade descriptors in mind.

Question	Answer	Marks
Section A		
1(a)(i)	Variable name, data type and use **one** mark, max **two** Several correct answers, the names chosen must be meaningful. Variables must relate to task 2 Example Name `totalMilk` Data type integer/real Use to store the total volume of the milk for the week (to the nearest whole litre) (1) Name `weeklyAverage` Data type integer/real Use to store the average yield per week (1)	**2**
1(a)(ii)	**One** mark per bullet point. • Data structure(s) given (1) • Data type (1) • Sample data (1) • More than one data structure described (1) Example A real array for each milking and an array of strings for the identity codes. There would be 14 arrays for the milking e.g. `mondayMorning, mondayEvening` Sample data for a cow could be 123, 23.5, 22.7	**4**
1(b)	Entering/selecting the identity code (1) method to ensure it is not a duplicate (1) Example Enter new identity code number Check if already in the list of code numbers	**2**

Question	Answer	Marks
1(c)	Any **five** from: 1 Initialisation for total weekly volume 2 loop control 3 calculation of running total for yield 4 calculation of average yield 5 output total and average yield per week with message outside loop 6 value(s) rounded Sample answer	**5**

```
total ← 0
FOR counter ← 1 TO numCows
    total ← total + mondayMorning(counter)
    total ← total + mondayEvening(counter)
    total ← total + tuesdayMorning(counter)
    total ← total + tuesdayEvening (counter)
    total ← total + wednesdayMorning(counter)
    total ← total + wednesdayEvening (counter)
    total ← total + thursdayMorning(counter)
    total ← total + thursdayEvening (counter)
    total ← total + fridayMorning(counter)
    total ← total + fridayEvening (counter)
    total ← total + saturdayMorning(counter)
    total ← total + saturdayEvening (counter)
    total ← total + sundayMorning(counter)
    total ← total + sundayEvening(counter)
NEXT counter
Average ← ROUND(total/numCows)
OUTPUT "Total volume of milk for week ", ROUND(total)
OUTPUT "Average weekly yield ", average
```

Question	Answer	Marks
1(d)(i)	Explanation Any **five** from: 1 Check each cow 2 Initialise day counter to zero 3 Check every day of the week 4 If daily yield is less than 12 … 5 … add one to day counter 6 If day counter >= 4 … 7 … identify/output identity code number(s)	**5**
1(d)(ii)	Explanation • Add new storage space to store code numbers for example new array/table/list • Add extra code to store these values if the condition was met	**2**

Question	Answer	Marks
Section B		
2(a)	**One** mark per correct pair of actions, process, Input/Output, Tests (apart from START and END) max 3 **One** mark complete Flowlines, **one** mark working flowlines, **one** mark correct use flowchart symbols 	**6**
2(b)	Any **two** from: • Use another counter/variable • Update this counter/variable when the number is less than zero/count all numbers **and** subtract the positive numbers • Output this counter/variable at the end // Output both counters at the end	**2**

Question	Answer	Marks
3(a)		3

Number1	Number2	Sign	Answer	OUTPUT
5	7	+	12	12
6	2	-	4	4
4	3	*	12	12
7	8	?	0	
0	0	/	(0)	

← 1 mark → ← 1 mark → ← 1 mark →

Question	Answer	Marks
3(b)	`CASE Sign OF … ENDCASE` (1) List +, -, *, / with correct assignments (1) `OTHERWISE Answer ← 0` (1) Example	3

```
CASE Sign OF
    '+' : Answer ← Number1 + Number2
    '-' : Answer ← Number1 - Number2
    '*' : Answer ← Number1 * Number2
    '/' : Answer ← Number1 / Number2
    OTHERWISE Answer ← 0
ENDCASE
```

Question	Answer	Marks
4(a)	Max **4** in total Any **3** from: • To ensure no changes are made on input / accuracy of transcription • Because the details do not have fixed, values or lengths to validate • Because there is no clear set of rules that can be used for validation Any **3** from: • The programmer could ask the contributor to type in each detail twice … • … and then check that both values are equal • … If they are not equal then the input should be rejected • The programmer could ask the contributor to check the details on the screen … • … and confirm that they are correct / same as the original • … or change them	**4**
4(b)	**One** mark for email and **one** mark for password Email – check for @ / format check / no spaces /valid characters // presence check // length check (not more than 254 characters) // uniqueness check Password – length check / numbers and letters etc. // uniqueness check not been used before // presence check	**2**

Question	Answer	Marks
5	**One** mark per value and reason, max **3** Example 1.00 – boundary rejected//rejected (underweight) // out of range(1) 1.02 – normal // valid // accepted weight in range (1) 1.10 – abnormal // erroneous // invalid // rejected (overweight) (1)	**3**

Question	Answer	Marks
6(a)	Fields 5	**1**
6(b)	**One** mark description of new code that will allow more than 1000 values **One** mark for example matching candidate's description Example Use a new character instead of N TT345	**2**
6(c)	see below	**4**

6(c)

Field:	At Risk	Age in Years	Type	Map Position
Table:	TREES	TREES	TREES	TREES
Sort:				
Show:	☐	☐	☑	☑
Criteria:	True	>100		
or:				

One mark per correct column

Cambridge Assessment International Education
Cambridge International General Certificate of Secondary Education

---

COMPUTER SCIENCE **0478/23**

Paper 2 **May/June 2018**

MARK SCHEME

Maximum Mark: 50

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 May/June 2018 series for most Cambridge IGCSE™, Cambridge International A and AS Level and Cambridge Pre-U components, and some Cambridge O Level components.

This syllabus is approved for use in England, Wales and Northern Ireland as a Cambridge International Level 1/Level 2 Certificate.

This document consists of **8** printed pages.

[Turn over

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
Section A		

1(a)(i)	**One** mark for any meaningful array name related to Task 1 × 2 e.g.
	`Tickets`
	`EntryTime`
	One mark for correct data type **AND** use related to Task 1 × 2 e.g.
	… integer to store the ticket numbers
	… real to store the entry times
		4

1(a)(ii)	**One** mark for any meaningful name for a constant **AND** value related to Task 3 × 2 e.g.
	`MaxTime`
	`8`
	`MaxFine`
	`100`
	One mark for correct use related to Task 3 × 2 e.g.
	…to store the maximum number of hours allowed in the car park
	…to store the current value of the fine for staying too long (in $)
		4

Question	Answer	Marks
1(b)	Any **six** from: • Initialisation of ticket numbers and arrays • Loop for day's transactions • Check for entry • Output of free spaces before entry • Suitable input prompts to select on entry • Relevant Inputs (to get ticket and to input time) • Recording of entry time and ticket number in arrays • Generate next ticket number • Update and display number of car park spaces available after entry	**6**

```
Ticket_Number ← 1 //This number would not reset each day
Ticket_Array[1:100]
Entry_Time[1:100]
Spaces ← 100
Count ← 1 //This number would reset each day

WHILE Count > 1 DO
    OUTPUT "Available Spaces" Spaces
    INPUT "Press enter to get ticket", Entry
    OUTPUT "Next Ticket Number", Ticket_Number
    INPUT "Current time", Current_Time
    Ticket_Array[Count] ←  Ticket_Number
    Entry_Time[Count] ←  Current_Time
    Ticket_Number  ←  Ticket_Number + 1
    Spaces ← Spaces – 1
    Count ← Count + 1
ENDWHILE
```

Question	Answer	Marks
1(c)	Any **four** from: • Explanation of how the checking of the length of stay was done • Correct comparison to check length of stay against maximum • Explanation of how the extra charge is calculated • Correct calculation to work out the regular parking charge • Explanation of suitable output to show parking charge and extra charge if appropriate	**4**
1(d)	**One** mark for each correct test data item and related reason for **Task 1** (Answers **MUST** relate to pre-release task) e.g. Test data: –102 Reason: To check that negative values for ticket numbers are rejected Test data: 85 Reason: To check that normal ticket number data is accepted	**2**

Question	Answer	Marks
	Section B	
2(a)	**One** mark for description **one** mark for example e.g. To test if the data entered is possible / reasonable … … A range check tests that data entered fits within specified values. Allow any correct validation check as an example	**2**
2(b)	**One** mark for description **one** mark for example e.g. To test if the data input is the same as the data that was intended to be input … … A double entry check expects each item of data to be entered twice and compares both entries to check they are the same. Allow any correct verification check as an example	**2**

Question	Answer	Marks

3	**One** mark for each correct answer
	`Counter = 0`
	`FOR Count = 1 TO 30`
	`Total = Total + Number`
	`NEXT Count`
		4

Question	Answer	Marks
4(a)	Any **two** from: • Expects a number to be input • Checks if the number is greater than 100 • Outputs the result of the test • Specific output example	**2**
4(b)(i)	**One** mark for correct answer e.g. Use a (condition controlled) loop	**1**
4(b)(ii)	**One** mark for each point • Initialisation of Number variable • Correct loop statements • Correct INPUT and OUTPUT e.g.	**3**

```
INPUT Number
WHILE Number > 100 DO
  OUTPUT "The number is too large"
  INPUT Number
ENDWHILE
OUTPUT "The number is acceptable"
```

or

```
INPUT Number
REPEAT
  IF Number > 100
    THEN
      OUTPUT "The number is too large"
  ENDIF
  INPUT Number
UNTIL Number <= 100
OUTPUT "The number is acceptable"
```

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>5(a)</td><td>
<table>
<tr><th>Max</th><th>Counter</th><th>Num</th><th>OUTPUT</th></tr>
<tr><td>–1000.00</td><td>0</td><td>6.30</td><td></td></tr>
<tr><td>6.30</td><td>1</td><td>18.62</td><td></td></tr>
<tr><td>18.62</td><td>2</td><td>50.01</td><td></td></tr>
<tr><td>50.01</td><td>3</td><td>3.13</td><td></td></tr>
<tr><td>50.01</td><td>4</td><td>2.05</td><td></td></tr>
<tr><td>50.01</td><td>5</td><td>50.10</td><td></td></tr>
<tr><td>50.10</td><td>6</td><td>40.35</td><td></td></tr>
<tr><td>50.10</td><td>7</td><td>30.69</td><td></td></tr>
<tr><td>50.10</td><td>8</td><td>0.85</td><td></td></tr>
<tr><td>50.10</td><td>9</td><td>17.30</td><td></td></tr>
<tr><td>50.10</td><td>10</td><td></td><td>50.10</td></tr>
<tr><td></td><td></td><td></td><td></td></tr>
</table>
← 1 mark →← 1 mark →← 1 mark →
</td><td>**3**</td></tr>
<tr><td>5(b)</td><td>**One** mark for each correct change (max **two**)
Box 2 Change the initialization value of the current ‘Max’ variable to a very high number
Box 4 Change the inequality from > to <
Boxes 2, 4, 5, 8 Change the Max variable to something more suitable e.g. Min</td><td>**2**</td></tr>
</table>

Question	Answer	Marks
6(a)	**One** mark for correct answer 20	**1**
6(b)(i)	**One** mark for correct answer CatNo	**1**
6(b)(ii)	**One** mark for correct answer It is a unique identifier	**1**

Question	Answer	Marks
6(c)	**One** mark for every **two** correct data types Field: CatNo – Data Type: Text Field: Title – Data Type: Text Field: Genre 1 – Data Type: Text Field: Stream – Data Type: Boolean / Text	**2**
6(d)	**One** mark for each correct row 18m02 Golfwatch Yes No Yes 18m03 Chair 27 Yes Yes No Accept if drawn in a table, but don't allow any punctuation	**2**
6(e)	See query grid below. **One** mark per completely correct column / group of columns as shown.	**4**

6(c):

Field	Data Type
CatNo	Text
Title	Text
Genre 1	Text
Stream	Boolean / Text

6(e):

Field:	CatNo	Title	Genre 1	Genre 2	Stream
Table:	2018MOV	2018MOV	2018MOV	2018MOV	2018MOV
Sort:		Ascending			
Show:	☑	☑	☐	☐	☐
Criteria:			="Sci-Fi"		Yes
or:				="Sci-Fi"	Yes
	← 1 mark	→	← 1 mark →	← 1 mark →	← 1 mark →

One mark per completely correct column / group of columns as shown.

Cambridge International Examinations
Cambridge International General Certificate of Secondary Education

---

COMPUTER SCIENCE **0478/21**

Paper 2 Problem-solving and Programming **May/June 2018**

PRE-RELEASE MATERIAL

No Additional Materials are required.

This material should be given to the relevant teachers and candidates as soon as it has been received at the Centre.

READ THESE INSTRUCTIONS FIRST

Candidates should use this material in preparation for the examination. Candidates should attempt the practical programming tasks using their chosen high-level, procedural programming language.

This syllabus is approved for use in England, Wales and Northern Ireland as a Cambridge International Level 1/Level 2 Certificate.

This document consists of **2** printed pages.

DC (NF) 148542/1

[Turn over

In preparation for the examination candidates should attempt the following practical tasks by **writing and testing a program or programs**.

A computer shop will build a computer from components to meet a customer's requirements. For each request for a computer to be built, an estimate of the cost is produced. The component stock level is checked; if all the components are in stock, a firm order to build the computer can be placed. A program is required to work out the cost of the computer, update the stock levels and provide a daily summary of orders for the shop owner.

Write and test a program or programs for the computer shop owner.

- Your program or programs must include appropriate prompts for the entry of data.
- Error messages and other output need to be set out clearly and understandably.
- All variables, constants and other identifiers must have meaningful names.

You will need to complete these **three** tasks. Each task must be fully tested.

TASK 1 – Produce an estimate.

Write a program for TASK 1 to calculate the cost of building a computer using these components:

Component	**Choices**	**Prices in $**
Processor	p3 / p5 / p7	100 / 120 / 200
RAM	16 GB / 32 GB	75 / 150
Storage	1 TB / 2 TB	50 / 100
Screen	19" / 23"	65 / 120
Case	Mini Tower / Midi Tower	40 / 70
USB ports	2 ports / 4 ports	10 / 20

The customer makes a choice for each component and an estimate is produced. The estimate must show a unique estimate number, the components chosen and the price of each component. The estimate must also show the total cost of the computer, which is calculated as the sum of the cost of the components chosen plus 20%.

TASK 2 – Place an order.

Using your estimate from TASK 1, check if the components required are in stock. If all the components are in stock then update the stock levels. Add the unique estimate number to the list of order numbers. Add the customer's details and today's date to the estimate details to finalise the order. Print two copies of the order, one for the customer and one for the shop.

TASK 3 –Summarise the day's orders.

Extend TASK 2 to provide an end of day summary showing the number of orders made, the total number of each component sold and the value of the orders.

Cambridge International Examinations
Cambridge International General Certificate of Secondary Education

COMPUTER SCIENCE **0478/22**

Paper 2 Problem-solving and Programming **May/June 2018**

PRE-RELEASE MATERIAL

No Additional Materials are required.

This material should be given to the relevant teachers and candidates as soon as it has been received at the Centre.

READ THESE INSTRUCTIONS FIRST

Candidates should use this material in preparation for the examination. Candidates should attempt the practical programming tasks using their chosen high-level, procedural programming language.

This document consists of **2** printed pages.

DC (NH) 148632/2

In preparation for the examination candidates should attempt the following practical tasks by **writing and testing a program or programs**.

A farmer records the milk production of a herd of cows. Every cow has a unique 3-digit identity code. Each cow can be milked twice a day, seven days a week. The volume of milk from each cow is recorded in litres correct to one decimal place (yield) every time the cow is milked. The size of the herd is fixed. At the end of the week the total and the average yield for each cow for that week is calculated.

The farmer identifies the cow that has produced the most milk that week. The farmer also identifies any cows that have produced less than 12 litres of milk on four or more days that week.

A program is required to record the yield for each cow every time it is milked, calculate the total weekly volume of milk for the herd and the average yield per cow in a week. The program must also identify the cow with the best yield that week and identify any cows with a yield of less than 12 litres of milk for four or more days that week.

Write and test a program or programs for the farmer.

- Your program or programs must include appropriate prompts for the entry of data.
- Error messages and other output need to be set out clearly and understandably.
- All variables, constants and other identifiers must have meaningful names.

You will need to complete these **three** tasks. Each task must be fully tested.

TASK 1 – Record the yield.

Write a program for TASK 1 to record the milk yields for a week. The program records and stores the identity code number and the yield every time a cow is milked.

TASK 2 – Calculate the statistics.

Using your recorded data from TASK 1, calculate and display the total weekly volume of milk for the herd to the nearest whole litre. Calculate and display the average yield per cow in a week to the nearest whole litre.

TASK 3 – Identify the most productive cow and cows that are producing a low volume of milk.

Extend TASK 2 to identify and display the identity code number and weekly yield of the cow that has produced the most milk. Also identify and display the identity code numbers of any cows with a yield of less than 12 litres of milk for four days or more in the week.

Cambridge International Examinations
Cambridge International General Certificate of Secondary Education

---

COMPUTER SCIENCE **0478/23**

Paper 2 Problem-solving and Programming **May/June 2018**

PRE-RELEASE MATERIAL

No Additional Materials are required.

This material should be given to the relevant teachers and candidates as soon as it has been received at the Centre.

READ THESE INSTRUCTIONS FIRST

Candidates should use this material in preparation for the examination. Candidates should attempt the practical programming tasks using their chosen high-level, procedural programming language.

This document consists of **2** printed pages.

DC (NH) 148634/3

In preparation for the examination candidates should attempt the following practical tasks by **writing and testing a program or programs**.

A car park has space for 100 cars and a barrier entrance and exit system. There is a display at the entrance to show how many spaces are empty. Cars are issued a ticket with a unique number on entry and the time of issue is stored. The car park charges $1.50 per hour and the fee is paid at a machine before leaving the car park. At the machine, the ticket number and departure time are entered; the fee is calculated by the machine and the amount due is paid by the ticket holder. Cars cannot stay overnight; the system is reset at midnight.

Write and test a program or programs for the car park manager.

- Your program or programs must include appropriate prompts for the entry of data.
- Error messages and other output need to be set out clearly and understandably.
- All variables, constants and other identifiers must have meaningful names.

You will need to complete these **three** tasks. Each task must be fully tested.

TASK 1 – Operating the car park.

The system is reset at midnight every day.
Set up a system using arrays and with suitable prompts that will carry out the following as cars enter or leave the car park:

On Entry:
- display the number of empty car park spaces
- issue the next available ticket number
- store the current time and the ticket number
- display the updated number of empty car park spaces.

On Exit:
- input a ticket number and departure time
- output the amount of time the car stayed at the car park
- delete the ticket number from the array
- display the updated number of empty car park spaces.

TASK 2 – Working out the cost and daily takings.

Amend the program so that it will calculate the amount to be paid using a charge of $1.50 per hour, or part of an hour (i.e. any amount of time into the next hour is charged for a whole hour). The amount to be paid is displayed and is added to a running total for the day, before the ticket number is deleted from the array. At the end of the day, the following information is displayed:
- total daily takings
- number of cars that have used the car park
- average charge per car
- average length of stay per car.

TASK 3 – Introducing parking restrictions.

The car park manager decides to restrict the length of stay to a maximum of eight hours, and will charge an extra $100 if a car overstays. Modify your program to implement this change and ensure the driver is aware of this extra charge. Output the number of cars that have overstayed in a day.

Cambridge International Examinations
Cambridge International General Certificate of Secondary Education

CANDIDATE NAME	

CENTRE NUMBER		CANDIDATE NUMBER	

COMPUTER SCIENCE **0478/11**

Paper 1 Theory **May/June 2018**

1 hour 45 minutes

Candidates answer on the Question Paper.

No Additional Materials are required.

No calculators allowed.

READ THESE INSTRUCTIONS FIRST

Write your Centre number, candidate number and name in the spaces at the top of this page.
Write in dark blue or black pen.
You may use an HB pencil for any diagrams, graphs or rough working.
Do not use staples, paper clips, glue or correction fluid.
DO **NOT** WRITE IN ANY BARCODES.

Answer **all** questions.

No marks will be awarded for using brand names of software packages or hardware.

Any businesses described in this paper are entirely fictitious.

At the end of the examination, fasten all your work securely together.

The number of marks is given in brackets [ ] at the end of each question or part question.

The maximum number of marks is 75.

This syllabus is approved for use in England, Wales and Northern Ireland as a Cambridge International Level 1/Level 2 Certificate.

This document consists of **11** printed pages and **1** blank page.

DC (SR/SW) 163639/3 **R**

CAMBRIDGE
International Examinations

1 Jane answers an examination question about computers and data correctly.
Six different words or numbers have been removed from her answer.

Complete the sentences in Jane's answer, using the list given. Not all items in the list need to be used.

- 2
- 10
- 16
- analogue
- binary
- denary
- digital
- hexadecimal

As humans, we process …………………………………… data, but a computer cannot

process this type of data. For a computer to be able to process data it needs to be

converted to …………………………………… data.

As humans, we mostly use a …………………………………… number system;

this is a base …………………………………… number system.

Computers use a …………………………………… number system;

this is a base …………………………………… number system.

[6]

2 Dheeraj identifies **three** hexadecimal numbers.

Write the **denary** number for each of the three hexadecimal numbers:

2A ....

101 ....

21E ....

[3]

Working Space

....

....

....

....

....

3 The three binary numbers in the registers A, B and C have been transmitted from one computer to another.

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

One binary number has been transmitted incorrectly. This is identified through the use of a parity bit.

Identify which register contains the binary number that has been transmitted **incorrectly**. Explain the reason for your choice.

The binary number that has been transmitted incorrectly is in **Register** ....

Explanation ....

....

....

....

....

.... [4]

4 Michele wants to email a file to Elsa. The file is too large so it must be compressed.

(a) Name **two** types of compression that Michele could use.

Compression type 1 ....

Compression type 2 ....
[2]

(b) The file Michele is sending contains the source code for a large computer program.

Identify which type of compression would be most suitable for Michele to use.

Explain your choice.

Compression type..

Explanation...

....

....

....

....

....
[4]

5 **Six** components of the Von Neumann model for a computer system and **six** descriptions are given.

Draw a line to match each component to the most suitable description.

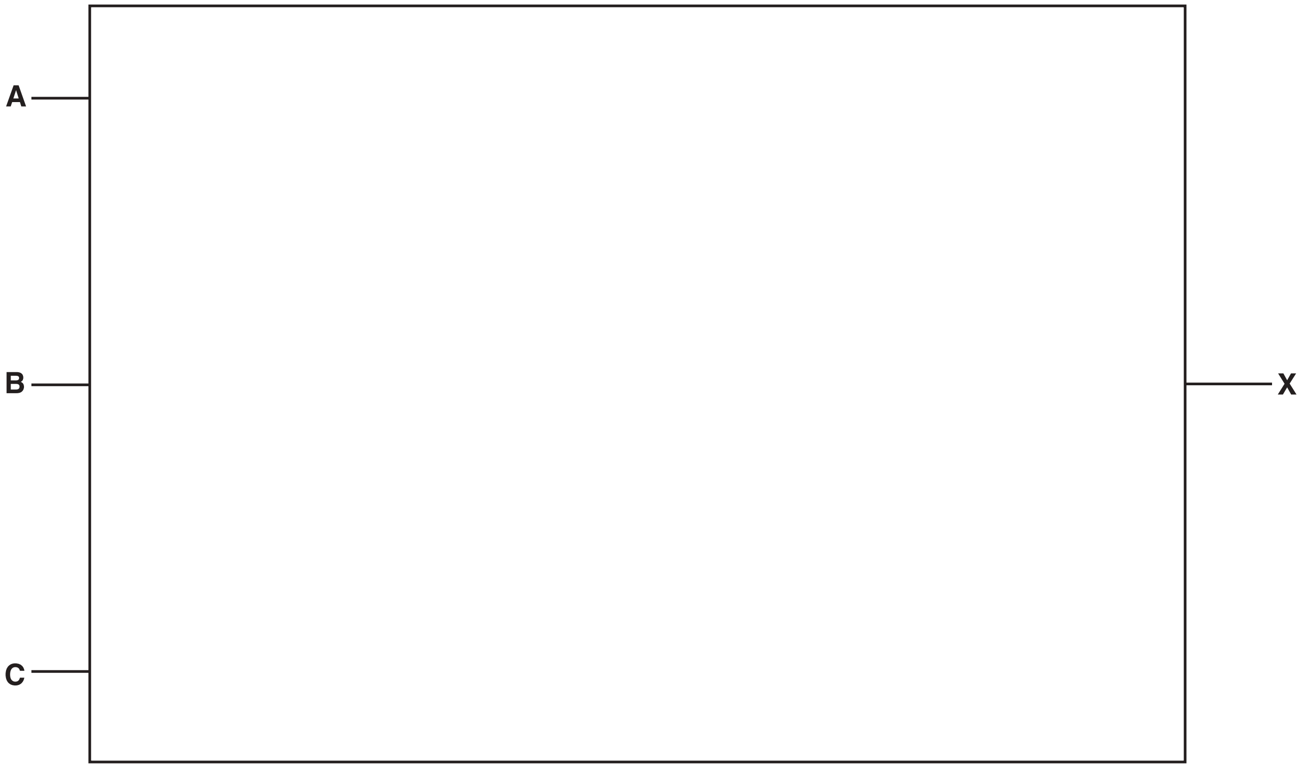
Component	**Description**
Immediate access store (IAS)	Holds data and instructions when they are loaded from main memory and are waiting to be processed.
Register	Holds data temporarily that is currently being used in a calculation.
Control unit (CU)	Holds data or instructions temporarily when they are being processed.
Accumulator (ACC)	Manages the flow of data and interaction between the components of the processor.
Arithmetic logic unit (ALU)	Carries out the calculations on data.
Bus	Pathway for transmitting data and instructions.

[5]

6 Consider the logic statement:

X = 1 if ((A is NOT 1 OR B is 1) NOR C is 1) NAND ((A is 1 AND C is 1) NOR B is 1)

(a) Draw a logic circuit to represent the given logic statement.



[6]

(b) Complete the truth table for the given logic statement.

A	B	C	Working space	X
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

[4]

7 Translators, such as a compiler and an interpreter, are used when writing and running computer programs.

Describe how a compiler and an interpreter translates a computer program.

Compiler ....

....

....

....

....

....

Interpreter ....

....

....

....

....

....
[6]

8 A supermarket uses a barcode scanner to read the barcodes on its products.

(a) Describe how the barcode scanner reads the barcode.

....

....

....

....

....

....

....

....
[4]

(b) Explain how the barcode system could help the supermarket manage its stock.

....

....

....

....

....

....
[3]

(c) An infrared touch screen is used to view and navigate the supermarket stock system.

Explain how the infrared touch screen detects a user's touch.

....

....

....

....

....

....

....

....
[4]

(d) The supermarket uses secondary storage and off-line storage to store data about its stock.

Explain what is meant by secondary storage and off-line storage.

Secondary storage ....

....

....

....

Off-line storage ....

....

....

....

[4]

9 A business wants to use a biometric security system to control entry to the office.

The system will use a biometric device and a microprocessor.

Explain how the biometric security system will make use of the biometric device and the microprocessor to control entry to the office.

....

....

....

....

....

....

....

....

....

....

....

....

[6]

10 RockICT is a music business that has a website to allow customers to view and buy the products it sells.

The website consists of web pages.

(a) Describe what is meant by HTML structure and presentation for a web page.

....

....

....

....

....

....

....

....[4]

(b) The URL for the music company's website is:

$$\text{https://www.}\underbrace{\text{rockict.net/}}_{\textbf{Part 1}}\underbrace{\text{index.htm}}_{\textbf{Part 2}}$$

(i) Identify what **Part 1** and **Part 2** represent in this URL.

Part 1 ....

Part 2 ....
[2]

(ii) Describe what is meant by **https**.

....

....

....

....[2]

(c) When a customer enters the website, a message is displayed:

"RockICT makes use of cookies. By continuing to browse you are agreeing to our use of cookies."

Explain why the music company uses cookies.

....

....

....

.... [2]

(d) The music company is concerned about the security of its website.

The company uses a proxy server as part of its security system.

Describe the role of a proxy server in the security system.

....

....

....

....

....

....

....

.... [4]

BLANK PAGE

Cambridge International Examinations
Cambridge International General Certificate of Secondary Education

CANDIDATE NAME			
CENTRE NUMBER	☐☐☐☐☐	CANDIDATE NUMBER	☐☐☐☐

8003295921

COMPUTER SCIENCE **0478/12**

Paper 1 Theory **May/June 2018**

1 hour 45 minutes

Candidates answer on the Question Paper.

No Additional Materials are required.

No calculators allowed.

READ THESE INSTRUCTIONS FIRST

Write your Centre number, candidate number and name in the spaces at the top of this page.
Write in dark blue or black pen.
You may use an HB pencil for any diagrams, graphs or rough working.
Do not use staples, paper clips, glue or correction fluid.
DO **NOT** WRITE IN ANY BARCODES.

Answer **all** questions.
No marks will be awarded for using brand names of software packages or hardware.

Any businesses described in this paper are entirely fictitious.

At the end of the examination, fasten all your work securely together.
The number of marks is given in brackets [ ] at the end of each question or part question.

The maximum number of marks is 75.

This syllabus is approved for use in England, Wales and Northern Ireland as a Cambridge International Level 1/Level 2 Certificate.

This document consists of **11** printed pages and **1** blank page.

DC (SC/SW) 150229/2

CAMBRIDGE
International Examinations

[Turn over

1 Different units of data can be used to represent the size of a file, as it changes in size.

Fill in the missing units of data, using the list given:

- byte
- gigabyte (GB)
- megabyte (MB)
- nibble

The units of data increase in size from smallest to largest.

Smallest	bit
↓	....
	....
	kilobyte (kB)
	....
	....
Largest	terabyte (TB)

[4]

2 **(a)** Nancy has captured images of her holiday with her camera. The captured images are stored as digital photo files on her camera.

Explain how the captured images are converted to digital photo files.

....

....

....

....

....

....

....

.... [4]

(b) Nancy wants to email the photos to Nadia.

Many of the photos are very large files, so Nancy needs to reduce their file size as much as possible.

Identify which type of compression would be most suitable for Nancy to use. Explain your choice.

Compression type ....

Explanation ....

....

....

....

....

.... [4]

3 A stopwatch uses six digits to display hours, minutes and seconds.

The stopwatch is stopped at:

0 2	:	3 1	:	5 8
Hours		**Minutes**		**Seconds**

An 8-bit register is used to store each pair of digits.

(a) Write the 8-bit binary numbers that are currently stored for the **Hours**, **Minutes** and **Seconds**.

Hours								
Minutes								
Seconds								

[3]

(b) The stopwatch is started again and then stopped.

When the watch is stopped, the 8-bit binary registers show:

Hours	0	0	0	0	0	1	0	1
Minutes	0	0	0	1	1	0	1	0
Seconds	0	0	1	1	0	1	1	1

Write the denary values that will now be shown on the stopwatch.

	:		:	
Hours		**Minutes**		**Seconds**

[3]

4 Jafar is using the Internet when he gets the message:

"D03, page is not available"

Jafar remembers that hexadecimal is often used to represent binary values in error codes.

Convert the hexadecimal number in the error message into 12-bit binary.

[3]

5 The three binary numbers in the registers X, Y and Z have been transmitted from one computer to another.

								Parity bit
Register X	1	0	0	1	0	0	1	0
Register Y	1	1	1	0	0	1	1	1
Register Z	1	1	1	0	1	0	0	1

Only **one** binary number has been transmitted correctly. This is identified through the use of a parity bit.

Identify which register contains the binary number that has been transmitted **correctly**. Explain the reason for your choice.

The binary number that has been transmitted correctly is in **Register** ....

Explanation ....

....

....

....

....

....

[4]

6 Kelvin correctly answers an examination question about the Von Neumann model.

Eight different terms have been removed from his answer.

Complete the sentences in Kelvin's answer, using the list given.

Not all items in the list need to be used.

- accumulator (ACC)
- address bus
- arithmetic logic unit (ALU)
- control unit (CU)
- data bus
- executed
- fetches
- immediate access store (IAS)
- memory address register (MAR)
- memory data register (MDR)
- program counter (PC)
- saved
- transmits

The central processing unit (CPU) ....

the data and instructions needed and stores them in the

.... to wait to be processed.

The .... holds the address of the next

instruction. This address is sent to the ....

The data from this address is sent to the ....

The instruction can then be decoded and ....

Any calculations that are carried out on the data are done by the

.... During calculations, the data is temporarily

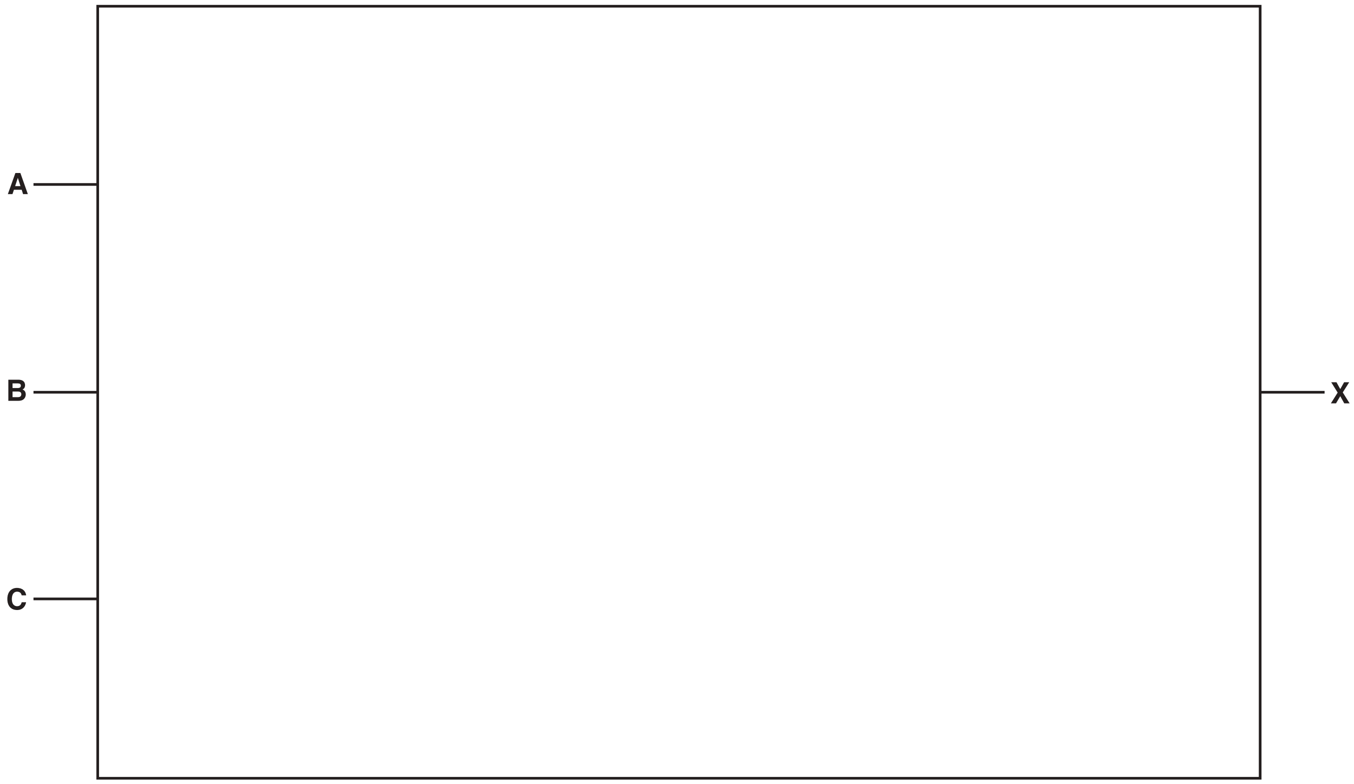
held in a register called the ....

[8]

7 Consider the logic statement:

X = 1 if ((A is 1 AND B is NOT 1) NAND C is 1) XOR ((A is 1 AND C is 1) OR B is 1)

(a) Draw a logic circuit to represent the given logic statement.



[6]

(b) Complete the truth table for the given logic statement.

A	B	C	Working space	X
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

[4]

8 Dimitri is writing a computer program in a high-level language.

He needs to send just the machine code for the program to his friend, electronically.

It is important that the program is executed as quickly as possible.

Identify which translator will be most suitable for Dimitri to use. Explain your choice.

Type of translator ....

Explanation ....

....

....

....

....

.... [4]

9 An advertisement in a magazine displays this barcode:

(a) Identify this type of barcode.

.... [1]

(b) Explain how the data stored in this barcode is read.

....

....

....

....

....

....

....

.... [4]

10 Alexandra has a new mobile device.

It has a touch screen that uses capacitive technology.

(a) Describe how a capacitive touch screen registers Alexandra's touch.

....

....

....

....

....

....

....

.... [4]

(b) Alexandra is wearing gloves because it is cold.

She presses an icon on her touch screen but her action is not registered.

(i) Explain why the touch screen will not register her touch.

....

....

....

.... [2]

(ii) Alexandra does not want to remove her gloves.

Explain how Alexandra could use her mobile device whilst still wearing gloves.

....

....

....

.... [2]

11 A factory uses a security system to control a security light. The system uses a sensor and a microprocessor.

Explain how the security system makes use of the sensor and the microprocessor to control the security light.

....

....

....

....

....

....

....

....

....

....

....

.... [6]

12 **(a)** Selma has some important personal information that she needs to email to her employer.

She wants to make sure that if the personal information is intercepted, it cannot be understood.

(i) State how Selma could email her personal data more securely.

.... [1]

(ii) Describe how your chosen solution works.

....

....

....

....

....

....

....

....

....

.... [5]

(b) Selma wants to make sure that the information received is correct.

A parity check can be used to detect errors.

Describe another error detection method that can be used to check the information received is correct.

Error detection method ....

Description ....

....

....

....

[3]

BLANK PAGE

Cambridge International Examinations
Cambridge International General Certificate of Secondary Education

CANDIDATE NAME	

CENTRE NUMBER		CANDIDATE NUMBER	

0557422193

COMPUTER SCIENCE **0478/13**

Paper 1 Theory **May/June 2018**

1 hour 45 minutes

Candidates answer on the Question Paper.

No Additional Materials are required.

No calculators allowed.

READ THESE INSTRUCTIONS FIRST

Write your Centre number, candidate number and name in the spaces at the top of this page.
Write in dark blue or black pen.
You may use an HB pencil for any diagrams, graphs or rough working.
Do not use staples, paper clips, glue or correction fluid.
DO **NOT** WRITE IN ANY BARCODES.

Answer **all** questions.
No marks will be awarded for using brand names of software packages or hardware.

Any businesses described in this paper are entirely fictitious.

At the end of the examination, fasten all your work securely together.
The number of marks is given in brackets [ ] at the end of each question or part question.

The maximum number of marks is 75.

This syllabus is approved for use in England, Wales and Northern Ireland as a Cambridge International Level 1/Level 2 Certificate.

This document consists of **11** printed pages and **1** blank page.

DC (SC/CGW) 148529/3

CAMBRIDGE
International Examinations

[Turn over

1 State **five** sensors that could be used in the following applications.

Give a **different** type of sensor for each application.

Application	**Sensor**
Weighing a baby in a hospital	
Turning off a kettle when the water boils	
Controlling an automatic door	
Monitoring the air quality in an aeroplane	
Counting cars crossing a bridge	

[5]

2 Draw a line to connect each term to the correct application.

Term	**Application**
Simplex	A telephone that can receive and transmit audio signals simultaneously.
Duplex	A two-way radio (walkie-talkie) that can receive and transmit messages, but not at the same time.
Half-duplex	A microphone that transmits data to a MIDI system.

[2]

3 Three security issues that could affect users online are **phishing**, **pharming** and **spam**.

Explain what is meant by each security issue.

Phishing ....

....

....

....

Pharming ....

....

....

....

Spam ....

....

....

....
[6]

4 A company transmits data to external storage at the end of each day.

(a) Parity checks can be used to check for errors during data transmission.

The system uses **odd parity**.

(i) Tick (✓) to show for each of the received bytes whether they have been **transmitted correctly** or **transmitted incorrectly**.

Received byte	**Transmitted correctly (✓)**	**Transmitted incorrectly (✓)**
10001011		
10101110		
01011101		
00100101		

[4]

(ii) State **one** other method that could be used to check for transmission errors.

.... [1]

(b) Data can be transferred using parallel or serial data transmission.

(i) Describe what is meant by parallel data transmission.

....

....

....

.... [2]

(ii) Give **one** application of parallel data transmission.

....

.... [1]

(iii) Explain why serial data transmission is normally used for transferring data over a long distance.

....

....

....

.... [2]

(c) Data transferred over a network is encrypted to improve data security.

The system uses 64-bit symmetric encryption.

(i) Explain how encryption improves data security.

....

....

....

.... [2]

(ii) Explain **one** method that could be used to increase the level of security provided by the encryption.

....

....

....

.... [2]

5 **(a)** Convert the denary number 107 to binary.

.... [1]

(b) Represent the denary number 300 as it would be stored in a 12-bit binary register.

.... [2]

(c) Convert the denary number 179 to hexadecimal.

.... [2]

6 One of the roles of an operating system is to deal with interrupts.

(a) Explain the term interrupt.

....

....

....

.... [2]

(b) Identify **three** devices that make use of interrupts.

Device 1 ....

Device 2 ....

Device 3 ....

[3]

7 A train station uses large touch screens to allow passengers to search for train information and buy tickets.

(a) State **three** benefits of using a touch screen in the train station.

Benefit 1 ....

....

Benefit 2 ....

....

Benefit 3 ....

....
[3]

(b) The touch screens at the station use resistive touch technology.

Describe how resistive touch technology works.

....

....

....

....

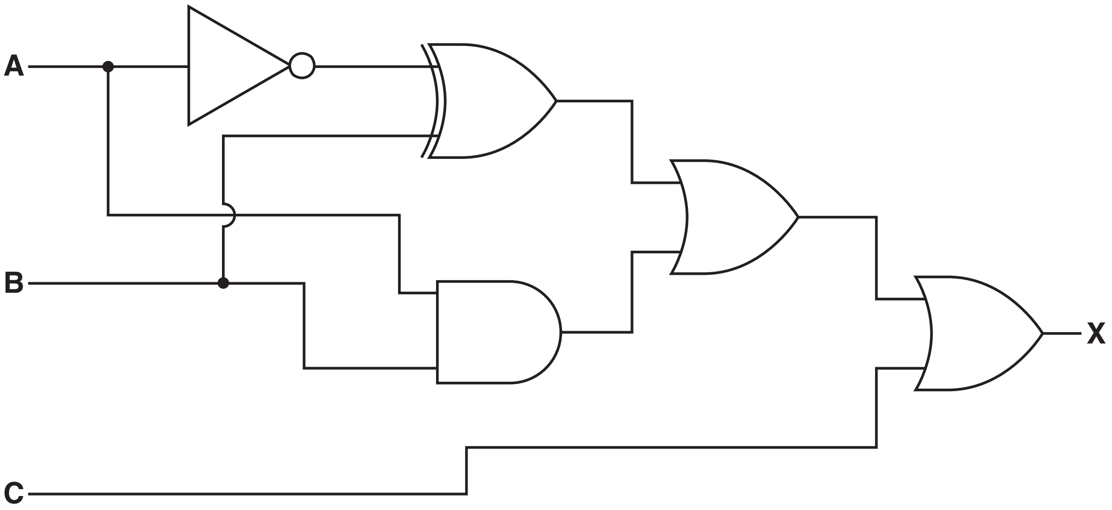
....

....

....

.... [4]

8 A logic circuit is shown below.



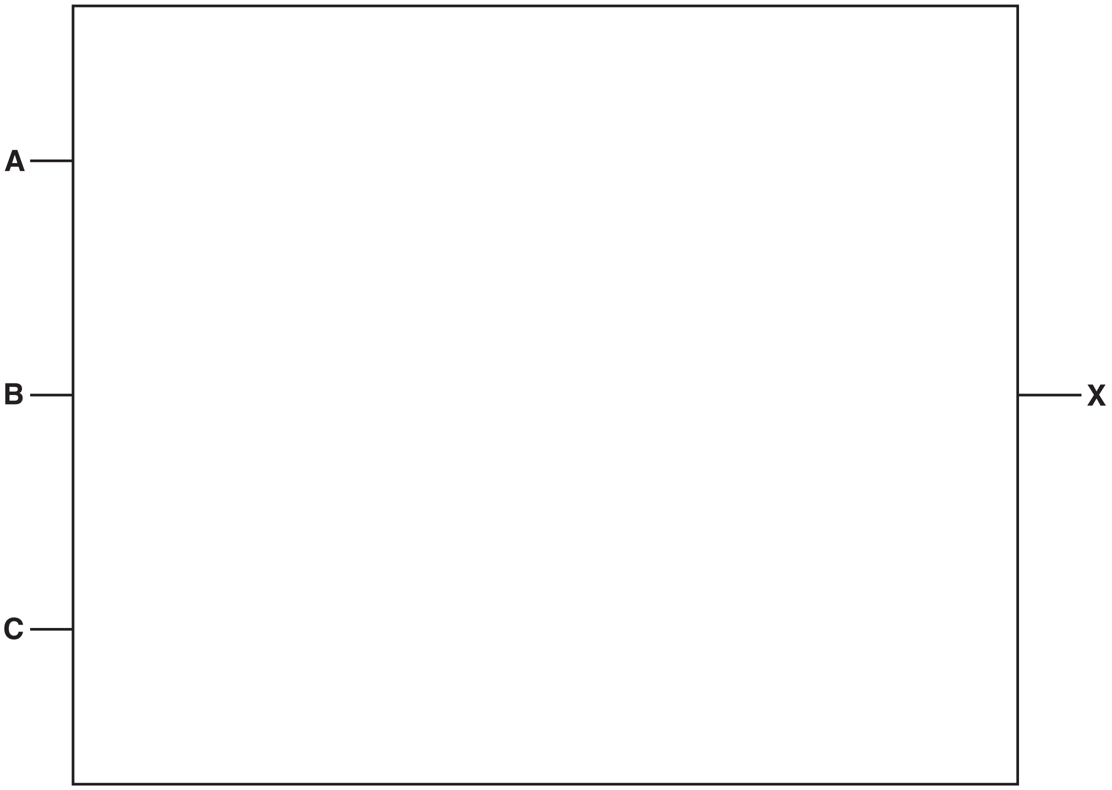
(a) Complete the truth table for the given logic circuit.

A	B	C	Working space	X
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

[4]

(b) Draw a logic circuit corresponding to this logic statement:

X = 1 if (A is NOT 1) OR ((B is 1 OR C is 1) AND (B is NOT 1 OR A is NOT 1))



[6]

9 Three types of translators are **assemblers**, **compilers** and **interpreters**.

Tick (✓) the appropriate boxes to show which statements apply to each type of translator.

Statement	**Assembler** **(✓)**	**Compiler** **(✓)**	**Interpreter** **(✓)**
Translates high-level language into machine code			
Provides error diagnostics			
Translates whole program to object code in one operation			
Translates and executes one line of code at a time			

[3]

10 Explain how an instruction is fetched in a computer based on the Von Neumann model.

....

....

....

....

....

....

....

....

....

....

....

.... [6]

11 Identify **three** similarities between CDs and DVDs.

1 ....

....

2 ....

....

3 ....

....

[3]

12 An image is to be stored electronically.

The image is 256 pixels high by 200 pixels wide with a 16-bit colour depth.

Calculate the file size of the image. **You must show all of your working.**

File size .... kB [3]

13 Describe the role of an Internet Service Provider (ISP).

....

....

....

....

....

....

....

....

....

....

....

.... [6]

BLANK PAGE

Cambridge International Examinations
Cambridge International General Certificate of Secondary Education

CANDIDATE NAME

CENTRE NUMBER

CANDIDATE NUMBER

1819492409

COMPUTER SCIENCE **0478/21**

Paper 2 Problem-solving and Programming **May/June 2018**

1 hour 45 minutes

Candidates answer on the Question Paper.

No Additional Materials are required.

No calculators allowed.

READ THESE INSTRUCTIONS FIRST

Write your Centre number, candidate number and name in the spaces at the top of this page.
Write in dark blue or black pen.
You may use an HB pencil for any diagrams, graphs or rough working.
Do not use staples, paper clips, glue or correction fluid.
DO **NOT** WRITE IN ANY BARCODES.

Answer **all** questions.

DO NOT ATTEMPT TASKS 1, 2 AND 3 in the pre-release material; these are for information only.

You are advised to spend no more than **40 minutes** on **Section A** (Question 1).

No marks will be awarded for using brand names of software packages or hardware.

At the end of the examination, fasten all your work securely together.

The number of marks is given in brackets [ ] at the end of each question or part question.

The maximum number of marks is 50.

This document consists of **11** printed pages and **1** blank page.

DC (NF) 148543/2

Section A

You are advised to spend no longer than 40 minutes answering this section.

Here is a copy of the pre-release material.

DO NOT attempt Tasks 1, 2 and 3 now.

Use the pre-release material and your experience from attempting the tasks before the examination to answer Question 1.

Pre-release material

A computer shop will build a computer from components to meet a customer's requirements. For each request for a computer to be built, an estimate of the cost is produced. The component stock level is checked; if all the components are in stock, a firm order to build the computer can be placed. A program is required to work out the cost of the computer, update the stock levels and provide a daily summary of orders for the shop owner.

Write and test a program or programs for the computer shop owner.

- Your program or programs must include appropriate prompts for the entry of data.
- Error messages and other output need to be set out clearly and understandably.
- All variables, constants and other identifiers must have meaningful names.

You will need to complete these **three** tasks. Each task must be fully tested.

TASK 1 – Produce an estimate.

Write a program for TASK 1 to calculate the cost of building a computer using these components.

Component	**Choices**	**Prices in $**
Processor	p3 / p5 / p7	100 / 120 / 200
RAM	16 GB / 32 GB	75 / 150
Storage	1 TB / 2 TB	50 / 100
Screen	19" / 23"	65 / 120
Case	Mini Tower / Midi Tower	40 / 70
USB ports	2 ports / 4 ports	10 / 20

The customer makes a choice for each component and an estimate is produced. The estimate must show a unique estimate number, the components chosen and the price of each component. The estimate must also show the total cost of the computer, which is calculated as the sum of the cost of the components chosen plus 20%.

TASK 2 – Place an order.

Using your estimate from TASK 1, check if the components required are in stock. If all the components are in stock then update the stock levels. Add the unique estimate number to the list of order numbers. Add the customer's details and today's date to the estimate details to finalise the order. Print two copies of the order, one for the customer and one for the shop.

TASK 3 – Summarise the day's orders.

Extend TASK 2 to provide an end of day summary showing the number of orders made, the total number of each component sold and the value of the orders.

1 **(a)** All variables, constants and other identifiers should have meaningful names.

(i) You recorded information for the estimate of the cost of building a computer in **Task 1**. Give a data structure that you created for **Task 1**, its name, data type and use.

Data structure ....

Name ....

Data type ....

Use ....

.... [4]

(ii) Describe the data structures that you have used in **Task 2** to record the customer details. Include sample data in the description.

....

....

....

....

....

....

....

.... [4]

(b) Explain how your program for **Task 1** produces a unique estimate number.

....

....

....

.... [2]

(c) Write an algorithm for part of **Task 2** to check that the chosen **processor** and chosen **RAM** are in stock, using **either** pseudocode, programming statements **or** a flowchart. Assume that **Task 1** has been completed. Do **not** check the other components or produce the order.

.... [5]

(d) Explain how your program completes **Task 3**. Any programming statements used in your answer must be fully explained.

.... [5]

Section B

2 **(a)** Write an algorithm to input 1000 numbers. Count how many numbers are positive and how many numbers are zero. Then output the results. Use **either** pseudocode **or** a flowchart.

.... [6]

(b) Give one change you could make to your algorithm to ensure initial testing is more manageable.

....

.... [1]

Question 3 starts on page 8.

3 The global trade item number (GTIN-8) barcode has seven digits and a check digit.
This pseudocode algorithm inputs seven digits and calculates the eighth digit, then outputs the GTIN-8.
`DIV(X,Y)`, finds the number of divides in division for example **`DIV(23,10)`** is 2.
`MOD(X,Y)`, finds the remainder in division for example **`MOD(23,10)`** is 3.

```
FOR Count ← 1 TO 7
   INPUT Number
   Digit(Count) ← Number
NEXT
Sum ← (Digit(1)+Digit(3)+Digit(5)+Digit(7))*3+Digit(2)+Digit(4)+Digit(6)
IF MOD(Sum,10) <> 0
   THEN Digit(8) ← DIV(Sum,10)*10 + 10 - Sum
   ELSE Digit(8) ← 0
ENDIF
OUTPUT "GTIN-8"
FOR Count ← 1 TO 8
   OUTPUT Digit(Count)
NEXT
```

(a) Complete the trace table for the input data: 5, 7, 0, 1, 2, 3, 4

Digit(1)	Digit(2)	Digit(3)	Digit(4)	Digit(5)	Digit(6)	Digit(7)	Digit(8)	Sum	OUTPUT

Complete the trace table for the input data: 4, 3, 1, 0, 2, 3, 1

Digit(1)	Digit(2)	Digit(3)	Digit(4)	Digit(5)	Digit(6)	Digit(7)	Digit(8)	Sum	OUTPUT

[5]

(b) Explain how you would change the algorithm to input eight digits (seven digits and the check digit) and output if the check digit entered is correct or not.

....

....

....

....

....

....

.... [3]

Question 4 starts on page 10.

4 A programmer has written a routine to check that prices are below $10.00. These values are used as test data.

10.00 9.99 ten

Explain why each value was chosen.

10.00 ....

....

....

9.99 ....

....

....

ten ....

....

....

[3]

5 Explain the difference between the programming concepts of **counting** and **totalling**.
Include an example of a programming statement for each concept in your explanation.

....

....

....

....

....

....

....

....

.... [4]

6 A database table, PERFORMANCE, is used to keep a record of the performances at a local theatre.

Show Number	Type	Title	Date	Sold Out
SN091	Comedy	An Evening at Home	01 Sept	Yes
SN102	Drama	Old Places	02 Oct	No
SN113	Jazz	Acoustic Evening	03 Nov	No
SN124	Classical	Mozart Evening	04 Dec	Yes
SN021	Classical	Bach Favourites	01 Feb	Yes
SN032	Jazz	30 Years of Jazz	02 Mar	Yes
SN043	Comedy	Street Night	03 Apr	No
SN054	Comedy	Hoot	04 May	No

(a) State the number of fields and records in the table.

Fields ....

Records ....
[2]

(b) Give **two** validation checks that could be performed on the **Show Number** field.

Validation check 1 ....

....

Validation check 2 ....

....
[2]

(c) Using the query-by-example grid, write a query to identify jazz performances that are not sold out. Only display the date and the title.

Field:					
Table:					
Sort:					
Show:	☐	☐	☐	☐	☐
Criteria:					
or:					

[4]

BLANK PAGE

Cambridge IGCSE

Cambridge International Examinations
Cambridge International General Certificate of Secondary Education

CANDIDATE NAME	

CENTRE NUMBER		CANDIDATE NUMBER	

7115246253

COMPUTER SCIENCE **0478/22**

Paper 2 Problem-solving and Programming **May/June 2018**

1 hour 45 minutes

Candidates answer on the Question Paper.

No Additional Materials are required.

No calculators allowed.

READ THESE INSTRUCTIONS FIRST

Write your Centre number, candidate number and name in the spaces at the top of this page.
Write in dark blue or black pen.
You may use an HB pencil for any diagrams, graphs or rough working.
Do not use staples, paper clips, glue or correction fluid.
DO **NOT** WRITE IN ANY BARCODES.

Answer **all** questions.

DO NOT ATTEMPT TASKS 1, 2 AND 3 in the pre-release material; these are for information only.

You are advised to spend no more than **40 minutes** on **Section A** (Question 1).

No marks will be awarded for using brand names of software packages or hardware.

At the end of the examination, fasten all your work securely together.

The number of marks is given in brackets [ ] at the end of each question or part question.

The maximum number of marks is 50.

This syllabus is approved for use in England, Wales and Northern Ireland as a Cambridge International Level 1/Level 2 Certificate.

This document consists of **11** printed pages and **1** blank page.

DC (NH) 148633/2

CAMBRIDGE
International Examinations

[Turn over

Section A

You are advised to spend no longer than 40 minutes answering this section.

Here is a copy of the pre-release material.

DO NOT attempt Tasks 1, 2 and 3 now.

Use the pre-release material and your experience from attempting the tasks before the examination to answer Question 1.

Pre-release material

A farmer records the milk production of a herd of cows. Every cow has a unique 3-digit identity code. Each cow can be milked twice a day, seven days a week. The volume of milk from each cow is recorded in litres correct to one decimal place (yield) every time the cow is milked. The size of the herd is fixed. At the end of the week the total and the average yield for each cow for that week is calculated.

The farmer identifies the cow that has produced the most milk that week. The farmer also identifies any cows that have produced less than 12 litres of milk on four or more days that week.

A program is required to record the yield for each cow every time it is milked, calculate the total weekly volume of milk for the herd and the average yield per cow in a week. The program must also identify the cow with the best yield that week and identify any cows with a yield of less than 12 litres of milk for four or more days that week.

Write and test a program or programs for the farmer.

- Your program or programs must include appropriate prompts for the entry of data.
- Error messages and other output need to be set out clearly and understandably.
- All variables, constants and other identifiers must have meaningful names.

You will need to complete these **three** tasks. Each task must be fully tested.

TASK 1 – Record the yield.

Write a program for TASK 1 to record the milk yields for a week. The program records and stores the identity code number and the yield every time a cow is milked.

TASK 2 – Calculate the statistics.

Using your recorded data from TASK 1, calculate and display the total weekly volume of milk for the herd to the nearest whole litre. Calculate and display the average yield per cow in a week to the nearest whole litre.

TASK 3 – Identify the most productive cow and cows that are producing a low volume of milk.

Extend TASK 2 to identify and display the identity code number and weekly yield of the cow that has produced the most milk. Also identify and display the identity code numbers of any cows with a yield of less than 12 litres of milk for four days or more in the week.

1 **(a)** All variables, constants and other identifiers should have meaningful names.

(i) State the name, the data type and the use of **two** variables that you have used in **Task 2**.

Variable 1 name ....

Data type ....

Use ....

Variable 2 name ....

Data type ....

Use ....

[2]

(ii) Describe, with the aid of some sample data, the data structures that you have used to record the data for the cows in **Task 1**.

....

....

....

....

....

....

....

....[4]

(b) Explain how your program for **Task 1** ensures that each 3-digit identity code is unique.

....

....

....

....[2]

(c) Write an algorithm for **Task 2**, using **either** pseudocode, programming statements **or** a flowchart**.** Assume that **Task 1** has been completed.

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....[5]

(d) **(i)** Explain how your program for **Task 3** finds the cows with a daily yield of less than 12 litres of milk for four days or more in the week. Any programming statements used in your answer must be fully explained.

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....[5]

(ii) Explain how you would extend your program for **Task 3** to **store** the identity code number(s) of those cows with a yield of less than 12 litres of milk for four days or more in the week.

....

....

....

....

....

....[2]

Section B

2 **(a)** Draw a flowchart for an algorithm to input numbers. Reject any numbers that are negative and count how many numbers are positive. When the number zero is input, the process ends and the count of positive numbers is output.

[6]

(b) Explain the changes you will make to your algorithm to also count the negative numbers.

....

....

....

....[2]

Question 3 starts on Page 8.

3 This pseudocode algorithm inputs two non-zero numbers and a sign, and then performs the calculation shown by the sign. An input of zero for the first number terminates the process.

```
INPUT Number1, Number2, Sign
WHILE Number1 <> 0
   IF Sign = '+' THEN Answer ← Number1 + Number2 ENDIF
   IF Sign = '-' THEN Answer ← Number1 - Number2 ENDIF
   IF Sign = '*' THEN Answer ← Number1 * Number2 ENDIF
   IF Sign = '/' THEN Answer ← Number1 / Number2 ENDIF
   IF Sign <> '/' AND Sign <> '*' AND Sign <> '-' AND Sign <> '+'
      THEN Answer ← 0
   ENDIF
   IF Answer <> 0 THEN OUTPUT Answer ENDIF
   INPUT Number1, Number2, Sign
ENDWHILE
```

(a) Complete the trace table for the input data:
5, 7, +, 6, 2, –, 4, 3, *, 7, 8, ?, 0, 0, /

Number1	**Number2**	**Sign**	**Answer**	**OUTPUT**

[3]

(b) Show how you could improve the algorithm written in pseudocode by writing an alternative type of conditional statement in pseudocode.

....

....

....

....

....

....

....

....

....

....[3]

4 A programmer has written a routine to store the name, email address and password of a contributor to a website's discussion group.

(a) The programmer has chosen to verify the name, email address and password.

Explain why verification was chosen and describe how the programmer would verify this data.

....

....

....

....

....

....

....

....

....[4]

(b) The programmer has also decided to validate the email address and the password.

Describe validation checks that could be used.

Email address ....

....

....

Password ....

....

....

[2]

5 A program checks that the weight of a basket of fruit is over 1.00 kilograms and under 1.10 kilograms. Weights are recorded to an accuracy of two decimal places and any weight not in this form has already been rejected.

Give **three** weights as test data and for each weight state a reason for choosing it. All your reasons must be different.

Weight 1 ....

Reason...

....

Weight 2 ....

Reason...

....

Weight 3 ....

Reason...

....
[3]

6 A database table, TREES, is used to keep a record of the trees in a park. Each tree is given a unique number and is examined to see if it is at risk of dying. There are over 900 trees; part of the database table is shown.

Tree Number	Type	Map Position	Age in Years	At Risk
TN091	Acacia	A7	250	Y
TN172	Olive	C5	110	N
TN913	Cedar	B9	8	N
TN824	Banyan	A3	50	Y
TN021	Pine	D5	560	Y
TN532	Teak	C8	76	Y
TN043	Yew	B1	340	N
TN354	Spruce	D4	65	N
TN731	Elm	B10	22	Y
TN869	Oak	C9	13	N
TN954	Pine	E11	3	N

(a) State the number of fields in the table.

....[1]

(b) The tree numbering system uses TN followed by three digits. The numbering system will not work if there are over 1000 trees.
Describe, with the aid of an example, how you could change the tree numbering system to allow for over 1000 trees. Existing tree numbers must not be changed.

....

....

....

....[2]

(c) Using the query-by-example grid, write a query to identify at risk trees over 100 years old. Display only the type and the position on the map.

Field:					
Table:					
Sort:					
Show:	☐	☐	☐	☐	☐
Criteria:					
or:					

[4]

BLANK PAGE

Cambridge International Examinations
Cambridge International General Certificate of Secondary Education

CANDIDATE NAME			
CENTRE NUMBER		CANDIDATE NUMBER	

0574821219

COMPUTER SCIENCE **0478/23**

Paper 2 Problem-solving and Programming **May/June 2018**

1 hour 45 minutes

Candidates answer on the Question Paper.

No Additional Materials are required.

No calculators allowed.

READ THESE INSTRUCTIONS FIRST

Write your Centre number, candidate number and name in the spaces at the top of this page.
Write in dark blue or black pen.
You may use an HB pencil for any diagrams, graphs or rough working.
Do not use staples, paper clips, glue or correction fluid.
DO **NOT** WRITE IN ANY BARCODES.

Answer **all** questions.

DO NOT ATTEMPT TASKS 1, 2 AND 3 in the pre-release material; these are for information only.

You are advised to spend no more than **40 minutes** on **Section A** (Question 1).

No marks will be awarded for using brand names of software packages or hardware.

Any businesses described in this paper are entirely fictitious.

At the end of the examination, fasten all your work securely together.

The number of marks is given in brackets [ ] at the end of each question or part question.

The maximum number of marks is 50.

This syllabus is approved for use in England, Wales and Northern Ireland as a Cambridge International Level 1/Level 2 Certificate.

This document consists of **11** printed pages and **1** blank page.

DC (NH/FC) 148635/3

CAMBRIDGE
International Examinations

[Turn over

Section A

You are advised to spend no longer than 40 minutes answering this section.

Here is a copy of the pre-release material.

DO NOT attempt Tasks 1, 2 and 3 now.

Use the pre-release material and your experience from attempting the tasks before the examination to answer Question 1.

Pre-release material

A car park has space for 100 cars and a barrier entrance and exit system. There is a display at the entrance to show how many spaces are empty. Cars are issued a ticket with a unique number on entry and the time of issue is stored. The car park charges $1.50 per hour and the fee is paid at a machine before leaving the car park. At the machine, the ticket number and departure time are entered; the fee is calculated by the machine and the amount due is paid by the ticket holder. Cars cannot stay overnight; the system is reset at midnight.

Write and test a program or programs for the car park manager.

- Your program or programs must include appropriate prompts for the entry of data.
- Error messages and other output need to be set out clearly and understandably.
- All variables, constants and other identifiers must have meaningful names.

You will need to complete these **three** tasks. Each task must be fully tested.

TASK 1 – Operating the car park.

The system is reset at midnight every day.
Set up a system using arrays and with suitable prompts that will carry out the following as cars enter or leave the car park:

On Entry:
- display the number of empty car park spaces
- issue the next available ticket number
- store the current time and the ticket number
- display the updated number of empty car park spaces.

On Exit:
- input a ticket number and departure time
- output the amount of time the car stayed at the car park
- delete the ticket number from the array
- display the updated number of empty car park spaces.

TASK 2 – Working out the cost and daily takings.

Amend the program so that it will calculate the amount to be paid using a charge of $1.50 per hour, or part of an hour (i.e. any amount of time into the next hour is charged for a whole hour). The amount to be paid is displayed and is added to a running total for the day, before the ticket number is deleted from the array. At the end of the day, the following information is displayed:
- total daily takings
- number of cars that have used the car park
- average charge per car
- average length of stay per car.

TASK 3 – Introducing parking restrictions.

The car park manager decides to restrict the length of stay to a maximum of eight hours, and will charge an extra $100 if a car overstays. Modify your program to implement this change and ensure the driver is aware of this extra charge. Output the number of cars that have overstayed in a day.

1 **(a)** All variables, constants and other identifiers should have meaningful names.

(i) State the name, data type and use of **two** arrays you created for **Task 1**.

Array 1 name ....

Data type ....

Use ....

....

Array 2 name ....

Data type ....

Use ....

....
[4]

(ii) State the name, value and use of **two** constants you could have created for **Task 3**.

Constant 1 name..

Value ....

Use ....

....

Constant 2 name..

Value ....

Use ....

....
[4]

(b) Write an algorithm to perform the set up and 'On Entry' part of **Task 1**, using **either** pseudocode, programming statements **or** a flowchart.

....[6]

(c) Explain how your program calculates if a car has overstayed the permitted parking time and how the charge is calculated and output (part of **Task 3**). Any programming statements you use in your answer must be fully explained.

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....[4]

(d) One of the inputs required 'On Exit' in **Task 1** is ticket number. State **two** items of suitable test data you could use to test your input validation and state why you chose them.

Test data 1 ....

Reason ....

....

Test data 2 ....

Reason ....

....

[2]

Section B

2 Describe, using an example, the purpose of the following checks during data entry.

(a) Validation check

....

....

....[2]

(b) Verification check

....

....

....[2]

3 This section of program code reads the contents of the array, totals the numbers and prints out the sum and average of the numbers. Assume the array is full.

Complete the **four** missing items by writing them in the spaces provided in this code.

```
Numbers[1:30]

Total = 0

............................................ = 0

FOR Count = 1 TO ......................................................

Number = Numbers[Count]

Total = ...................................................... + Number

Counter = Counter + 1

............................................................................................ Count

PRINT 'The sum of the numbers you entered is ', Number

PRINT 'The average of the numbers you entered is ', Number / Counter
```

[4]

4 An algorithm is written in pseudocode:

```
INPUT Number
IF Number > 100
   THEN OUTPUT "The number is too large"
   ELSE OUTPUT "The number is acceptable"
ENDIF
```

(a) Describe the purpose of the algorithm.

....

....

....

....

....[2]

(b) (i) The algorithm only allows one attempt at inputting an acceptable value.

State how you would change the algorithm so that it continues until a suitable input is supplied.

....

....[1]

(ii) Re-write the algorithm in full, using pseudocode, to implement your answer to **part (b)(i).**

....

....

....

....

....

....

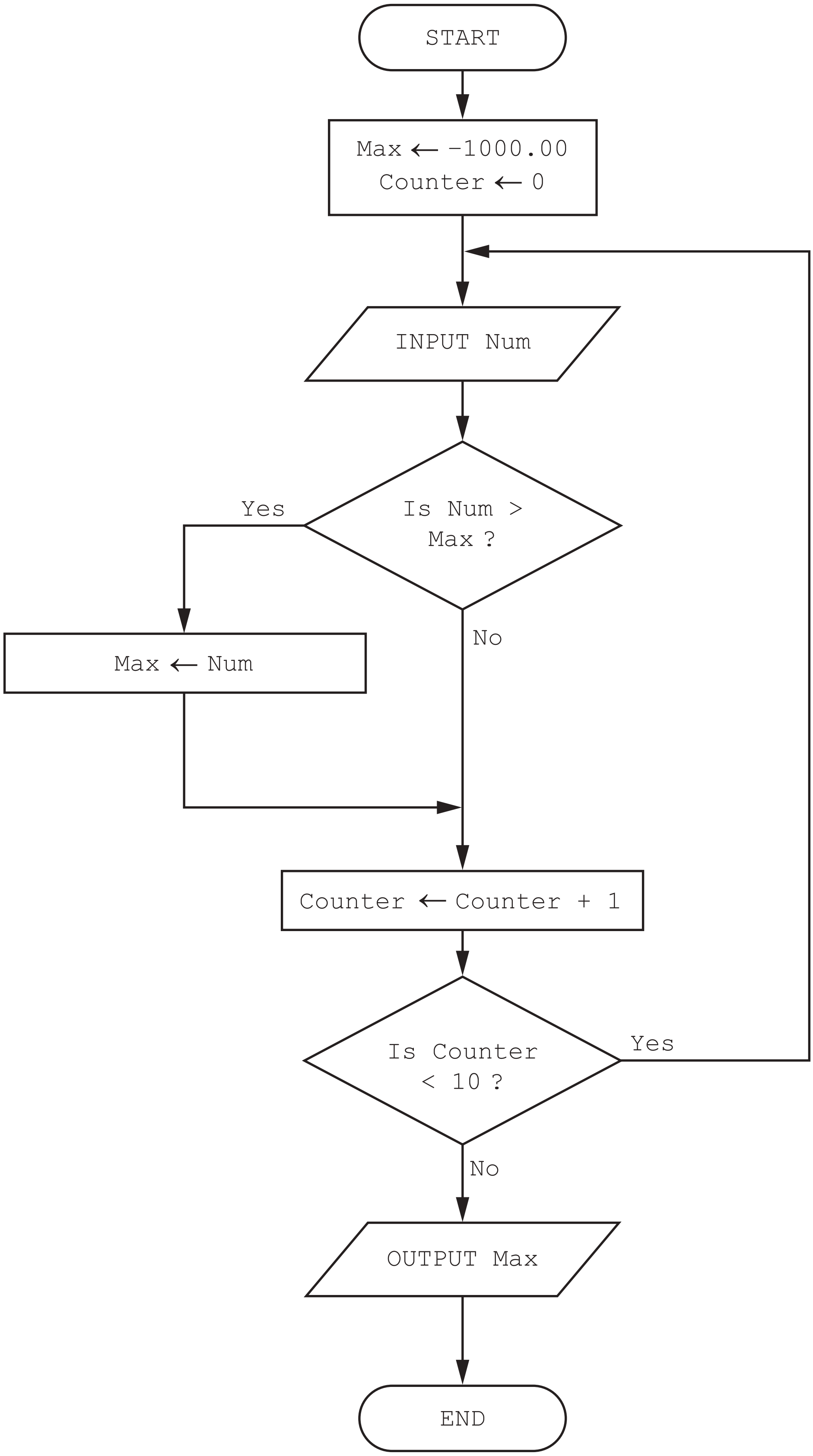
....

....

....

....[3]

5 The flowchart allows a set of 10 numbers to be entered; it finds and outputs the largest of these numbers.



(a) Complete the trace table for the input data:

6.30, 18.62, 50.01, 3.13, 2.05, 50.10, 40.35, 30.69, 0.85, 17.30

Max	**Counter**	**Num**	**OUTPUT**

[3]

(b) Describe **two** different changes you should make to the flowchart to find the smallest number instead of the largest number.

Change 1 ....

....

Change 2 ....

....

[2]

6 A shop that sells copies of movies to the public has set up a new database table called 2018MOV to store some new releases. Part of this table is given, showing the catalogue number, title, genres and available formats (Blu-ray, DVD or streaming) of each movie.

CatNo	Title	Genre 1	Genre 2	Blu-ray	DVD	Stream
18m01	Battery Rangers	Adventure	Fantasy	Yes	No	Yes
18m02	Golfwatch	Comedy	Drama	Yes	No	Yes
18m03	Chair 27	Comedy	Drama	Yes	Yes	No
18m04	Wander Woman	Action	Fantasy	Yes	No	Yes
18m05	Justine League	Action	Fantasy	Yes	Yes	Yes
18m06	That	Horror	Thriller	Yes	Yes	No
18m07	Insect Dude	Action	Fantasy	No	Yes	No
18m08	Dover Beach	Action	History	No	Yes	No
18m12	Slow 25	Action	Thriller	No	Yes	No
18m15	Kongkers	Adventure	Fantasy	No	Yes	No
18m16	Transducers: The Last Night	Action	Sci-Fi	Yes	Yes	Yes
18m17	The Pale Tower	Fantasy	Sci-Fi	Yes	Yes	No
18m19	Bea and the Bute	Fantasy	Romance	Yes	Yes	Yes
18m21	The Daddy	Action	Fantasy	No	No	Yes
18m22	Planet Wars: Episode X	Sci-Fi	Action	Yes	No	Yes
18m23	Guardians of the Milky Way	Action	Sci-Fi	Yes	Yes	Yes
18m26	Odin	Horror	Sci-Fi	No	Yes	Yes
18m27	That	Fantasy	Sci-Fi	No	No	Yes
18m30	Underneath	Action	Horror	Yes	No	No
18m31	Debatable Me	Animation	Action	Yes	Yes	No

(a) State the number of records in this part of the table.

....[1]

(b) **(i)** Give the name of the field that should be used for the primary key.

....[1]

(ii) State the reason for choosing this field for the primary key.

....

....[1]

(c) Complete the table to show the most appropriate data type for each field based on the data shown in the table at the start of question 6.

Field	Data type
CatNo	
Title	
Genre 1	
Stream	

[2]

(d) List the output that would be given by this query-by-example.

Field:	CatNo	Title	Genre 1	Blu-ray	DVD	Stream	
Table:	2018MOV	2018MOV	2018MOV	2018MOV	2018MOV	2018MOV	
Sort:							
Show:	☑	☑	☐	☑	☑	☑	☐
Criteria:			="Comedy"				
or:							

....

....

....

....[2]

(e) Using the query-by-example grid, write a query to identify all the movies that are categorised as Sci-Fi and available to stream. Only display the catalogue number and title of the film, with the titles listed in alphabetical order.

Field:							
Table:							
Sort:							
Show:	☐	☐	☐	☐	☐	☐	☐
Criteria:							
or:							

[4]

BLANK PAGE