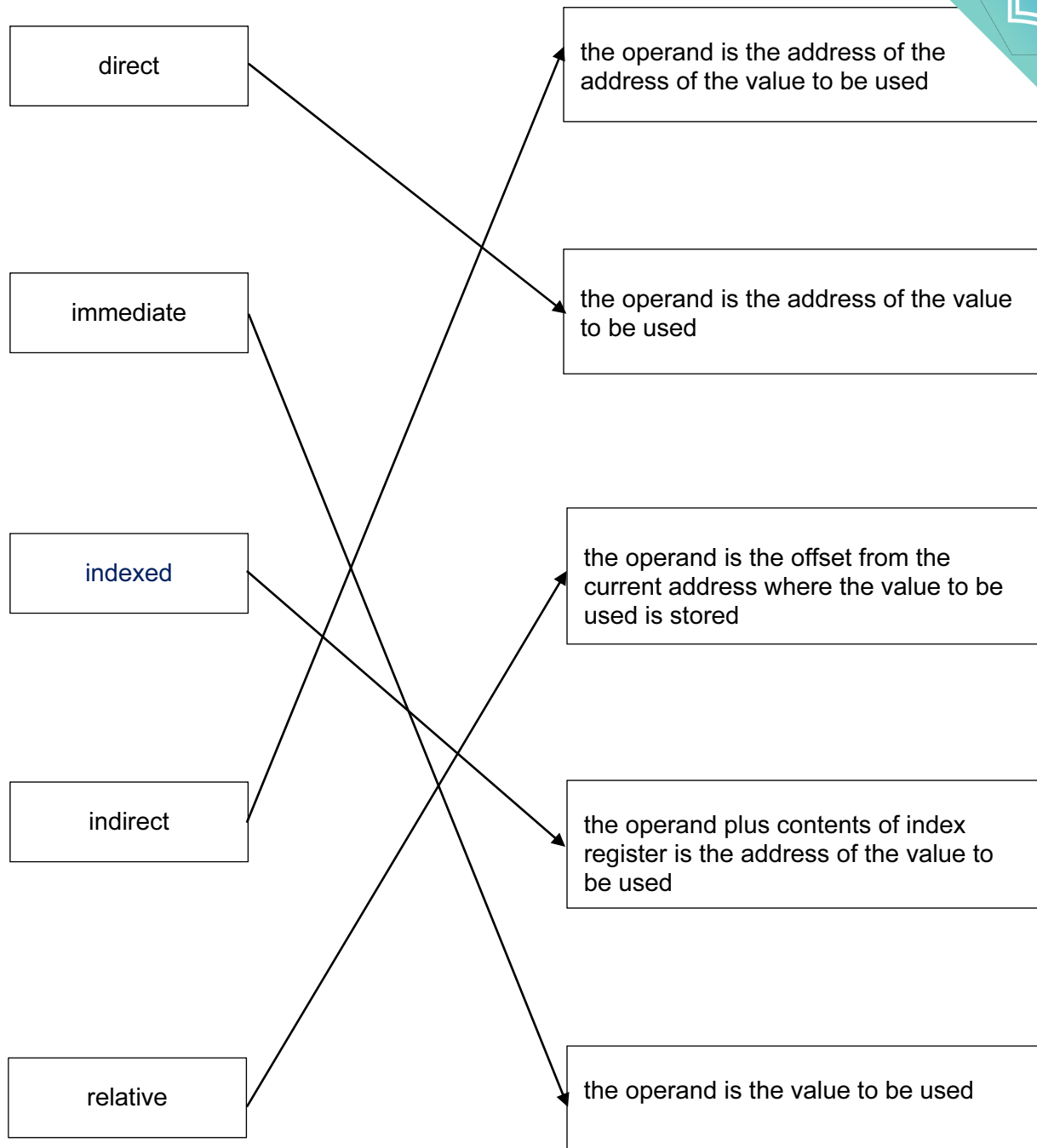


QUESTION 1.

3



[4]

QUESTION 2.



7 (a) (i)

Accumulator:

0	1	1	1	0	1	0	1
---	---	---	---	---	---	---	---

 [1]

(ii)

Accumulator:

0	1	1	0	1	0	0	1
---	---	---	---	---	---	---	---

 [1]

explanation

- content of 124 is 0 1 1 1 1 1 1 1
- this is equivalent to 127
- contents of 127 are 0 1 1 0 1 0 0 1

[2]

(iii)

Accumulator:

0	1	0	0	0	0	0	1
---	---	---	---	---	---	---	---

 [1]

explanation

- index register value = 6
- $120 + 6 = 126$
- contents of 126 placed in the accumulator

[2]



(b) 1 mark for each correct value in the table.

Accumulator	Memory address			
	320	321	322	323
	49	36	0	0
36				
37				
				37
49				
50				
			50	

[6]

QUESTION 3.



9 (a) (i) **One mark** for the contents of the accumulator and **one mark** for the reason. [2]

Accumulator contents: 0100 0101

Reason:

Address is 60

Contents of the index register is 8

And $60 + 8 = 68$ in denary gives the address

The contents of which is 0100 0101 in binary.

(ii) **0000 0111** [1]



(b)

Instruction address	Working space	ACC	Memory address				IX	C
			100	101	102	103		
			20	100	1	0	1	
50		20						
51		21						
52						21		
53		100						
54		120						
55								
56								
57								
59							2	
60		20						
61		120						
62								'x'
63								

One mark for each shaded block.

[7]

- Contents of the Accumulator in first 2 lines (instruction addresses 50 and 51)
- Updating address 103 (instruction 52)
- Loading the Accumulator and addition (instructions 53 and 54)
- Not executing instruction 58
- Incrementing the index register (instruction 59)
- Loading the Accumulator and addition (instructions 60 and 61)
- Correct output of 'x' (instruction 62)

QUESTION 4.

4 (a) 11001110

(b)

Instruction	Working space	ACC	Memory address				IX	OUTPUT
			90	91	92	93		
			2	90	55	34	2	
20		55						
21		54						
22			54					
23							3	
24		34						
25		33						
26								
27								
28								
31		67						
32						67		
33								'C'
34								

One mark each for:

- Instruction 20
- Instructions 21 and 22
- Instruction 23
- Instructions 24 and 25
- Not executing instructions 29 and 30
- Instructions 31 and 32
- Correct output



QUESTION 5.



8 (a) (i)

Accumulator:	1	0	0	1	0	1	1	1
--------------	---	---	---	---	---	---	---	---

[1]

(ii) **One mark** for answer and **two marks** for explanation

Accumulator:	1	1	0	0	0	0	1	0
--------------	---	---	---	---	---	---	---	---

- Index Register contains $1001 = 9$
- $800 + 9 = 809$

[3]

(b) (i) **ONE** mark for each correct row.

ACC	Memory address				OUTPUT
	800	801	802	803	
	40	50	0	90	
40					
90			90		
90			90		
					Z

[4]

(ii) **107**

[1]

Page 10	Mark Scheme
	Cambridge International AS/A Level – October/November 2016



(c) (i) Any **two** from:

- Only 128 / 256 characters can be represented
- Uses values 0 to 127 (or 255 if extended form) / one byte
- Many characters used in other languages cannot be represented
- In extended ASCII the characters from 128 to 255 may be coded differently in different systems

[2]

(ii) Any **two** from:

- Uses 16, 24 or 32 bits / two, three or four bytes
- Unicode is designed to be a superset of ASCII
- Designed so that most characters (in other languages) can be represented

[2]

QUESTION 6.

Cambridge International AS/A Level – October/November 2016



5 (a) (i)

0	1	1	1	0	0	1	1
---	---	---	---	---	---	---	---

(ii) **ONE** mark for Accumulator contents, **ONE** mark for the explanation.

1	0	1	1	0	0	0	1
---	---	---	---	---	---	---	---

- Index Register holds the value 4; $101 + 4 = 105$ so load data from address 105

[2]

(iii) **ONE** mark for Accumulator contents, **TWO** marks for the explanation.

0	1	0	0	1	0	1	1
---	---	---	---	---	---	---	---

- Memory address 103 contains the value 107
- So address 107 is the address from which to load the data

[3]

(b) **ONE** mark for each correct row.

ACC	Memory address			
	810	811	812	813
	28	41	0	0
28				
29				
			29	
41				
70				
				70

[6]

QUESTION 7.



4(a)(iv)	80	1
4(b)	00000001 00010001 00001100 01100001 Both correct op codes Operand 0001 0001 Operand 0110 0001	3 1 1 1
4(c)	256	1
4(d)(i)	07 C2 07 C2	2 1 1
4(d)(ii)	LDI 63 LDI 63	2 1 1



Question	Answer																																																																																	
5(a)(i)	- Count the number of one bits in the <u>first seven</u> bit positions - Add a 0 or 1 to bit position 0, to make the count of one bits an <u>odd</u> number																																																																																	
5(a)(ii)	A = 1 B = 1																																																																																	
5(a)(iii)	Two from: - A parity bit is worked out for each <u>column</u> 1 - The computer checks the parity of each bit position in parity byte // the computer generates copy of the parity byte and <u>compares</u> 1 - If incorrect parity then there is an error in the data received // No parity error means no error in the data received 1 - The position of the incorrect bit can be determined 1	2																																																																																
5(b)(i)	<table><tr><th colspan="8">Bit position</th></tr><tr><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td></tr></table> <table><tr><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td></tr></table>	Bit position								7	6	5	4	3	2	1	0	1	0	0	0	1	1	0	0	0	0	1	0	0	0	0	0	0	0	1	1	0	1	0	1	1	1	1	1	0	0	0	1	1	1	0	0	0	0	1	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	1	1	0	0	0	2
Bit position																																																																																		
7	6	5	4	3	2	1	0																																																																											
1	0	0	0	1	1	0	0																																																																											
0	0	1	0	0	0	0	0																																																																											
0	0	1	1	0	1	0	1																																																																											
1	1	1	1	0	0	0	1																																																																											
1	1	0	0	0	0	1	0																																																																											
0	0	1	0	0	1	0	0																																																																											
0	0	0	0	0	0	0	1																																																																											
0	1	0	1	1	0	0	0																																																																											
5(b)(ii)	Three from: - Consider each row in sequence 1 - Identify any row with incorrect parity 1 - Repeat the process for each column in sequence 1 - Identify where a row and column with incorrect parity intersect 1	3																																																																																



Question	Answer	
6(a)	Main memory management The user moves the mouse on the desktop Input/Output Management The user closes the Spreadsheet program Secondary storage Management The user selects the Save command to save their spreadsheet file Human computer interface The user selects the Print command to output their spreadsheet file One mark for each correct line from each left hand box to max <u>three</u> marks.	
6(b)(i)	File compression software	1
6(b)(ii)	Backup software	1
6(b)(iii)	Disk repair software	1
6(b)(iv)	Anti-virus software	1



Question	Answer	
7(a)	Two from: - The user's web browser is the client software 1 - The requested web page has program code / script embedded <u>within it</u> 1 - This code is interpreted by the web browser 1	
7(b)	Four from: - The browser parses the URL to obtain the Domain Name 1 - The browser software passes the Domain Name to the nearest Domain Name Server (DNS) 1 - The DNS stores a list of Domain Names and matching IP addresses 1 - The DNS Name Resolver looks for the Domain Name in its database 1 - If found the corresponding IP address is returned to the originator 1 - If not found the request is forwarded to another higher level DNS 1 - The original DNS adds the returned IP address to its cache 1 - The original DNS returns the IP address to the originator 1 - The browser uses the IP address to request the required web page from the <u>web server</u> 1 - The web server retrieves the page and delivers it to the originator 1 - The browser software interprets <u>the script</u> and displays the web page 1	Max 4
7(c)(i)	Message1, Message2 1 x 1	2
7(c)(ii)	6 – 19	1
7(c)(iii)	11	1
7(c)(iv)	Checks that the product code has not be left blank // presence check on product code	1
7(c)(v)	Two checks from: One mark for check and one mark for description - Range check 1 - Check the number entered is (say) between 1 and 100 1 - Format check 1 - Checks the product code is a particular format // Checks the number has digit characters only // by example 1 - Length check 1 - The number of items has exactly five characters 1 - Existence check 1 - To ensure the product code has been assigned 1	Max 4

QUESTION 8.



Question	Answer	Marks
5(a)(i)	351	1
5(a)(ii)	355	1
5(a)(iii)	22	1



Question	Answer																																															
5(a)(iv)	86																																															
5(b)	<table><thead><tr><th colspan="8">Op code</th><th colspan="8">Operand</th></tr></thead><tbody><tr><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td></tr></tbody></table> Both correct op codes 1 Operand 0100 0011 1 Operand 0000 0111 1	Op code								Operand								0	0	0	1	0	0	1	0	0	1	0	0	0	1	1	0	0	0	1	0	1	0	1	0	0	0	0	1	1	1	
Op code								Operand																																								
0	0	0	1	0	0	1	0	0	1	0	0	0	1	1																																		
0	0	0	1	0	1	0	1	0	0	0	0	1	1	1																																		
5(c)(i)	<table><tbody><tr><td>14</td><td>5E</td><td></td></tr><tr><td>14</td><td></td><td>1</td></tr><tr><td>5E</td><td></td><td>1</td></tr></tbody></table>	14	5E		14		1	5E		1	2																																					
14	5E																																															
14		1																																														
5E		1																																														
5(c)(ii)	<table><tbody><tr><td>LDR</td><td>#77</td><td></td></tr><tr><td>LDR</td><td></td><td>1</td></tr><tr><td>#77</td><td></td><td>1</td></tr></tbody></table>	LDR	#77		LDR		1	#77		1	2																																					
LDR	#77																																															
LDR		1																																														
#77		1																																														

QUESTION 9.



Cambridge Assessment International Education
Cambridge International Advanced Subsidiary and Advanced Level



COMPUTER SCIENCE

9608/11

Paper 1 Written Paper

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.

© IGCSE is a registered trademark.

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





Question	Answer									
1(a)(i)	119									
1(a)(ii)	−120									
1(a)(iii)	<table border="1"><tr><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>1</td><td>1</td><td>1</td></tr></table>	1	1	1	0	1	1	1	1	
1	1	1	0	1	1	1	1			
1(a)(iv)	Lowest value: −128 Highest value: +127	1								
1(b)(i)	0110 0101 0011	1								
1(b)(ii)	The second block of four binary digits represents a digit larger than 9 // 14	1								
1(b)(iii)	A string of digits on any electronic device displaying numeric values	1								

Question	Answer	Marks
2(a)		4
2(b)(i)	One mark from: - The program code can be translated to run on any processor / platform - Source code is translated into machine independent intermediate code not machine dependent code	1



Question	Answer
2(b)(ii)	Two marks from: - Java uses a two-step translation process - Java code is partially interpreted – partially compiled - Code is translated first into intermediate code / "bytecode"... ...using the Java compiler - The bytecode is finally interpreted by the Java Virtual Machine

Question	Answer	Marks																																													
3(a)	Two marks from: - Physical measures - Access rights - Encryption - Firewall - Use authentication methods such as usernames and passwords - Anti-malware program	Max 2																																													
3(b)(i)	<table border="1"><tr><td>7</td><td>X</td><td>6</td><td>=</td><td>42</td></tr><tr><td>8</td><td>X</td><td>5</td><td>=</td><td>40</td></tr><tr><td>6</td><td>X</td><td>4</td><td>=</td><td>24</td></tr><tr><td>5</td><td>X</td><td>3</td><td>=</td><td>15</td></tr><tr><td>3</td><td>X</td><td>2</td><td>=</td><td>6</td></tr><tr><td>1</td><td>X</td><td>1</td><td>=</td><td>1</td></tr><tr><td></td><td></td><td></td><td>Total:</td><td>128 / 11</td></tr><tr><td></td><td></td><td></td><td></td><td>11 R 7</td></tr><tr><td colspan="4">Check digit:</td><td>11 – 7 = 4</td></tr></table> 1 mark for 6 values 1 mark for 2 steps Accept 128 MOD 11 = 7 1 mark for subtraction Answer: 786531 4 (1 mark for answer)	7	X	6	=	42	8	X	5	=	40	6	X	4	=	24	5	X	3	=	15	3	X	2	=	6	1	X	1	=	1				Total:	128 / 11					11 R 7	Check digit:				11 – 7 = 4	4
7	X	6	=	42																																											
8	X	5	=	40																																											
6	X	4	=	24																																											
5	X	3	=	15																																											
3	X	2	=	6																																											
1	X	1	=	1																																											
			Total:	128 / 11																																											
				11 R 7																																											
Check digit:				11 – 7 = 4																																											
3(b)(ii)	One mark for name of check One mark for description Max two checks Uniqueness check Each PatientID must be unique Length check Each PatientID is exactly 7 characters Format check / Type check All 7 characters must be <u>digits</u> Presence check PatientID must be entered	Max 4																																													



Question	Answer																															
4(a)	A – System clock B – Control unit C – Main memory E – Control bus F – Data bus																															
4(b)	<table border="1"> <thead> <tr> <th>ACC</th><th>CountDown</th><th>OUTPUT</th></tr> </thead> <tbody> <tr> <td></td><td>15</td><td></td></tr> <tr> <td>67</td><td></td><td>C</td></tr> <tr> <td>15</td><td></td><td></td></tr> <tr> <td>14</td><td>14</td><td></td></tr> <tr> <td>51</td><td></td><td>3</td></tr> <tr> <td>14</td><td></td><td></td></tr> <tr> <td>13</td><td>13</td><td></td></tr> <tr> <td>32</td><td></td><td></td></tr> <tr> <td>88</td><td></td><td>x</td></tr> </tbody> </table> (1) (1) + (1) (1) (1)	ACC	CountDown	OUTPUT		15		67		C	15			14	14		51		3	14			13	13		32			88		x	5
ACC	CountDown	OUTPUT																														
	15																															
67		C																														
15																																
14	14																															
51		3																														
14																																
13	13																															
32																																
88		x																														
4(c)	Three marks from: - The assembler scans the assembly language instructions in sequence - When it meets a symbolic address checks to see if already in symbol table - If not, it adds it to the symbol table in the symbolic address column - If it is already in symbol table check if absolute address known - If the absolute address is known, it is entered in the appropriate cell - If the absolute address is not known mark / leave as unknown	Max 3																														
4(d)(i)	The op code / mnemonic / instruction table	1																														
4(d)(ii)	A – 1110 0110 0110 1000 (1) (1) B – E6 68 (1)	3																														



Question	Answer	
5(a)(i)	Three marks from: • Diaphragm / cone • (Voice) coil of wire • Spider / Suspension • (Permanent) Magnet • Basket • Dust cap • Outer frame	
5(a)(ii)	Four marks from: • Takes an electrical signal and translates it into physical vibrations to create sound waves • An electric current in the coil creates an electro-magnetic field • Changes in the audio signal causes the direction of the electric current to change • The direction of the current determines the polarity of the electro-magnet // changing the direction of the current changes the direction of the polarity of the electro-magnet • The electro-magnet is repelled by or attracted to the permanent magnet • Causing the coil to vibrate • The movement of the coil causes the cone / diaphragm to vibrate • That vibration is transmitted to the air in front of the cone / diaphragm as sound waves • The amount of movement will determine the frequency and amplitude of the sound wave produced	Max 4
5(b)(i)	One mark from: • External hard disk drive // SSD • External CD / DVD drive • Pen drive • Blu-ray drive	1
5(b)(ii)	Two marks from: • Additional secondary file storage // storing files • Backup of files • Archiving of files • Transfer files to second computer	Max 2



Question	Answer	
6(a)	Two marks from: - A system of moral principles - That guide behaviour / decision making - Based on philosophical / religious views - By example, e.g. respectful and considerate behaviour	
6(b)	One mark for identifying the issue One mark for correct principle One mark for possible action Max 2 issues (2 × 3 marks) 1 Uncomfortable with one of his colleagues Client and Employer // Management / Colleagues // Judgement // Self For example: Team building exercises // arranged meeting 2 Unfamiliar with programming language Self // Client and Employer // Product // Profession // Colleagues For example: Undergo training 3 Visit to unfamiliar workplace Client and employer // Management // Judgement // Profession // Colleagues For example: He should speak to his manager to discuss situation	Max 6

Question	Answer	Marks
7(a)(i)	<u>PatientID</u> } <u>DoctorID</u> } (1) <u>AppointmentDate, AppointmentTime</u> (1)	2
7(a)(ii)	<pre> graph TD PATIENT <--> DOCTOR PATIENT --> APPOINTMENT DOCTOR --> APPOINTMENT </pre> One PATIENT attends many APPOINTMENTs One DOCTOR takes many APPOINTMENTs Special case for 1 mark only (only if no one to many relationships shown) Many PATIENTs are seen by many DOCTORs	2



Question	Answer	
7(b)	Two marks from: Either: - Add an attribute (for example <code>Attended</code>) - To the appointment table // <code>APPOINTMENT</code> Or: - Add an attribute (for example <code>AppointmentsMissed</code>) - To the patient table // <code>PATIENT</code>	
7(c)(i)	Available to work at both <code>SITE-A</code> and <code>SITE-B</code>	1
7(c)(ii)	<code>APPOINTMENT(Site, AppointmentDate, AppointmentTime, DoctorID, PatientID)</code>	1
7(d)(i)	One mark per line <code>UPDATE DOCTOR</code> <code>SET DoctorID = '017'</code> <code>WHERE DoctorID = '117';</code>	3
7(d)(ii)	1 Mark per bullet, max 2 - Referential integrity should be maintained // Referential integrity could be violated. - Data becomes inconsistent - There may be records in the <code>APPOINTMENT</code> table showing doctor ID 117 - The <code>APPOINTMENT</code> table might not be automatically updated - Records in the <code>APPOINTMENT</code> table will become orphaned	Max 2
7(e)	One mark per line <code>SELECT AppointmentDate, AppointmentTime</code> <code>FROM APPOINTMENT</code> <code>WHERE PatientID = '556';</code>	3

QUESTION 10.



Question	Answer	
4(a)	1 Mark for each correct answer A – General purpose registers B – System clock C – ALU E – Control bus F – Address bus	
4(b)	1 Mark per bullet, max 2 - The clock sends out a number of pulses in a given time interval (clock speed) - Each processor instruction takes a certain number of clock cycles to execute - The higher the clock frequency, the shorter the execution time for the instruction // Increasing the clock frequency improves performance	2
4(c)(i)	1 Mark per bullet Maximum 2 for Macro Maximum 2 for Directive Maximum 3 in total Macro - A group of instructions given a name // subroutine - A group of instructions that need to be executed several times within the same program - The statements are written once and called using the name whenever they need to be executed - Macro code is inserted into the source file at each place it is called - By example Directive - An instruction that directs the assembler to do something - A directive is not a program instruction - It is information for the assembler - By example	3
4(c)(ii)	1 Mark for a suitable example For example: State the start address for the program //tell the assembler to set aside space for variables // include an external file etc.	1



Question	Answer																																							
4(d)	Mark as shown <table><thead><tr><th>ACC</th><th>Offset</th><th>OUTPUT</th></tr></thead><tbody><tr><td></td><td>10</td><td></td></tr><tr><td>50</td><td></td><td>2</td></tr><tr><td>10</td><td></td><td></td></tr><tr><td>11</td><td>11</td><td></td></tr><tr><td>65</td><td></td><td></td></tr><tr><td></td><td></td><td>A</td></tr><tr><td>11</td><td></td><td></td></tr><tr><td>12</td><td>12</td><td></td></tr><tr><td>89</td><td></td><td>Y</td></tr><tr><td>12</td><td></td><td></td></tr><tr><td>13</td><td>13</td><td></td></tr><tr><td>32</td><td></td><td></td></tr></tbody></table> 1 Mark for these two values, as first instructions 1 Mark for this value, in any row 1 Mark for this value, in any row 1 Mark for this value, after 65, nothing in between 1 Mark for the rest	ACC	Offset	OUTPUT		10		50		2	10			11	11		65					A	11			12	12		89		Y	12			13	13		32		
ACC	Offset	OUTPUT																																						
	10																																							
50		2																																						
10																																								
11	11																																							
65																																								
		A																																						
11																																								
12	12																																							
89		Y																																						
12																																								
13	13																																							
32																																								



Question	Answer										
4(e)	Mark as follows: Table entries: 1 Mark per bullet, max 4 • EndProg • 2 × Unknown • 9 • 14 • 8 Numbering: 1 Mark per bullet, max 2 • Relative address of Value is numbered 6 • Number given for EndProg is next number in sequence to relative address of Value • All numbers correct – award 2 marks <table border="1"> <thead> <tr> <th>Symbolic address</th><th>Relative address</th></tr> </thead> <tbody> <tr> <td><u>StartProg</u></td><td>0</td></tr> <tr> <td>Offset</td><td>UNKNOWN 9 (1)</td></tr> <tr> <td>Value</td><td>UNKNOWN (6) 14 (1)</td></tr> <tr> <td><u>EndProg</u> (7)</td><td>UNKNOWN (8) 8 (9)</td></tr> </tbody> </table>	Symbolic address	Relative address	<u>StartProg</u>	0	Offset	UNKNOWN 9 (1)	Value	UNKNOWN (6) 14 (1)	<u>EndProg</u> (7)	UNKNOWN (8) 8 (9)
Symbolic address	Relative address										
<u>StartProg</u>	0										
Offset	UNKNOWN 9 (1)										
Value	UNKNOWN (6) 14 (1)										
<u>EndProg</u> (7)	UNKNOWN (8) 8 (9)										

QUESTION 11.



Question	Answer	Marks
4(a)(i)	1 mark for each type of addressing Direct addressing - The operand is the address where the data is stored Indirect addressing - The operand is an address, that address holds another address where the data is stored	2



Question	Answer													
4(a)(ii)	1 mark per bullet point Direct addressing: 20 is the address of the data Indirect addressing: 20 is an address which holds the address where the data is stored													
4(b)	1 mark for 1 correct tick 2 marks for 3 correct ticks <table border="1" data-bbox="392 616 1240 880"> <thead> <tr> <th>Instruction</th><th>Symbolic</th><th>Absolute</th></tr> </thead> <tbody> <tr> <td>ADD 90</td><td></td><td>✓</td></tr> <tr> <td>CMP found</td><td>✓</td><td></td></tr> <tr> <td>STO 20</td><td></td><td>✓</td></tr> </tbody> </table>	Instruction	Symbolic	Absolute	ADD 90		✓	CMP found	✓		STO 20		✓	2
Instruction	Symbolic	Absolute												
ADD 90		✓												
CMP found	✓													
STO 20		✓												
4(c)(i)	186	1												
4(c)(ii)	BA	1												
4(c)(iii)	−70	1												
4(d)	1 mark per bullet point - Outputting * (instruction 71) - Storing 130 in 203 (instruction 72) - Loading, incrementing accumulator and storing in 204 (instructions 73, 74 and 75) - Incrementing Index Register (instruction 76) - Loading 133, comparing and jumping to 81 (instructions 77, 78 and 79) - Loading, comparing and jumping to 74 (instructions 81, 82 and 83) - Incrementing accumulator, storing in 204 and incrementing index register, loading 130 (instructions 74–79) - Outputting * to end (instructions 80–84)	8												

[illegible]

QUESTION 12.



Question	Answer	Marks
3(a)(i)	1 mark per bullet point Absolute addressing: - The operand is a numeric address // The numeric address is given // referring directly to a memory location Symbolic addressing: - The operand is a word/symbol // A word/symbol represents the memory location/address	2
3(a)(ii)	1 mark per example Absolute addressing: For example, <code>ADD 230</code> Symbolic addressing: For example, <code>ADD num1</code>	2
3(b)(i)	1 mark per bullet point Indexed addressing: - The address to be used is formed by: operand + the contents of the Index Register (IX) Immediate addressing: - The operand is not an address // the operand is the actual value to be loaded	2



Question	Answer	
3(b)(ii)	1 mark per example Indexed: For example, <code>LDX 20</code> Immediate: For example, <code>ADD #20</code>	
3(c)(i)	193	1
3(c)(ii)	C1	1
3(c)(iii)	–63	1
3(d)	1 mark per bullet point • Loading 2, comparing with 104 (instructions 40 and 41) • Loading 302 (instruction 43) • Comparing, and branching to 47 (instructions 44, 45) • Loading, decrementing accumulator and storing (instructions 47, 48 and 49) • Incrementing Index Register (instruction 50) • Loading 303, comparing and outputting + (instructions 43–46) • Loading, decrementing accumulator and storing, incrementing Index Register and end (instructions 47–51, 41, 42 and 54)	7

[illegible]

QUESTION 13.



Question	Answer	Marks
2(a)	1 mark per bullet point Relative addressing: • A number (the offset) is added to the base address to give the actual address Indexed addressing: • The contents of the index register are added to operand to give the actual address	2
2(b)(i)	242	1



Question	Answer																
2(b)(ii)	F2																
2(b)(iii)	−14																
2(b)(iv)	<table><tr><td>1</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td></tr></table>								1	1	1	1	0	0	1	1	1
1	1	1	1	0	0	1	1										
2(c)	1 mark per bullet point • Loading 8 (instruction 20) • Comparison and loading 453 (instructions 21–23) • Outputting & (instruction 26) • Loading, incrementing and storing in 96 (instructions 27–29) • Incrementing Index Register (instruction 30) • Jumping and loading 452 (instruction 23) • Jumping, loading, incrementing, storing in 96, incrementing IX and end (instructions 24–32)								7								



Instruction address	ACC	Memory address						IX	
		93	94	95	96	97	98		
		453	453	452	8	10	453	8	
20	8								
21									
22									
23	453								
24									
25									
26									&
27	8								

QUESTION 14.



Question	Answer		
5(a)	1 mark for two correct ticks, 2 marks for three correct ticks		
	Task	First pass	Second pass
	Creation of symbol table	✓	
	Expansion of macros	✓	
	Generation of object code		✓
	Removal of comments	✓	
5(b)	1 mark per bullet point to max 2 • Data movement • Input and output of data • Arithmetic operations • Jump instructions • Compare instructions		2
5(c)	1 mark per bullet point • Storing 0 in 300 (line 21) • Loading 65 (line 28) • Outputting A (line 29) • Loading 0 (line 30), incrementing ACC (line 31) and storing in 300 (line 32) • Incrementing IX (line 33) • Loading 67 (line 24) and adding 33 (line (25) • Outputting d (line 26) • Loading 1 (line 30), incrementing ACC (line 31), storing in 300 (line 32) and incrementing IX (line 33)		8

[illegible]

QUESTION 15.



Question	Answer	Marks
3(a)	1 mark for each error and correction • Line 02 should be +1 not -1 // $PC \leftarrow [PC] + 1$ • Line 03 should be double brackets around MAR // $MDR \leftarrow [[MAR]]$ • Line 04 should be MDR not MAR // $CIR \leftarrow [MDR]$	3
3(b)	1 mark for each group to max. 2 • Data movement • Arithmetic operations • (Unconditional and conditional) jump instructions • Compare instructions • Modes of addressing	2
3(c)	1 mark per bullet • Storing 0 in 401 (line 51) • Loading memory location 300, value 2 to ACC (line 52) • Adding 64 to ACC to give 66 (line 55) • Outputting B (line 56) • Load 0 (line 57), increment ACC (line 58) and store 1 in 401 (line 59) • Incrementing IX (line 60) • Loading 5 (line 52), adding 64 (line 55), outputting E (line 56) loading 1 (line 57), incrementing ACC (line 58), storing 2 in 401 (line 59) and incrementing IX (line 60) • Load 0 (line 52) <u>and end</u>	8



Question	Answer	
3(d)(i)	1 mark for correct answer 0100 0001	
3(d)(ii)	1 mark for correct answer 41	1
3(d)(iii)	1 mark for correct answer 0044	1

QUESTION 16.



Question	Answer	Marks																									
4(a)	1 mark per bullet - LDM #300 The (denary) number 300 is loaded (into the register) - LDD 300 The <u>contents</u> of address 300 are loaded (into the register)	2																									
4(b)	1 mark for JPN correct 1 mark for both of ADD and DEC correct 1 mark for LDR correct <table><tr><th></th><th>Description</th><th>Jump instruction</th><th>Arithmetic operation</th><th>Data movement</th></tr><tr><td>LDR #3</td><td>Load the number 3 to the Index Register</td><td></td><td></td><td>✓</td></tr><tr><td>ADD #2</td><td>Add 2 to the Accumulator</td><td></td><td>✓</td><td></td></tr><tr><td>JPN 22</td><td>Move to the instruction at address 22</td><td>✓</td><td></td><td></td></tr><tr><td>DEC ACC</td><td>Subtract 1 from the Accumulator</td><td></td><td>✓</td><td></td></tr></table>		Description	Jump instruction	Arithmetic operation	Data movement	LDR #3	Load the number 3 to the Index Register			✓	ADD #2	Add 2 to the Accumulator		✓		JPN 22	Move to the instruction at address 22	✓			DEC ACC	Subtract 1 from the Accumulator		✓		3
	Description	Jump instruction	Arithmetic operation	Data movement																							
LDR #3	Load the number 3 to the Index Register			✓																							
ADD #2	Add 2 to the Accumulator		✓																								
JPN 22	Move to the instruction at address 22	✓																									
DEC ACC	Subtract 1 from the Accumulator		✓																								



Question	Answer	
4(c)(i)	1 mark for hardware interrupt 1 mark for software interrupt For example: Hardware interrupt • Printer out of paper • No CD in drive Software interrupt • A running program needs input • Runtime error, e.g. division by zero	
4(c)(ii)	1 mark per bullet to max 5 • At the start / end of each fetch-execute cycle the processor checks for interrupt(s) • Check if an interrupt flag is set // Check if bit set in interrupt register • Processor identifies source of interrupt • Processor checks priority of interrupt • If interrupt priority is high enough // Lower priority interrupts are disabled • Processor saves current contents of registers // saves current job on stack • Processor calls interrupt handler / Interrupt Service Routine (ISR) • Address of ISR is loaded into Program Counter (PC) • When servicing of interrupt complete, processor restores registers // job from stack is restored • Lower priority interrupts are re-enabled • Processor continues with next F–E cycle	5

QUESTION 17.



Question	Answer	Marks
3(a)(i)	1 mark per table • CUSTOMER table has at least customer ID, customer name, address and contact details • ROOM has at least room number, room type, • BOOKING has at least booking ID, room number, customer ID, start date, number of nights CUSTOMER (<u>CustomerID</u>, Name, Address, ContactDetails) ROOM (RoomNumber, RoomType) BOOKING (BookingID, RoomNumber, CustomerID, StartDate, NumberNights)	3



Question	Answer		
3(a)(ii)	1 mark for 1 or 2 correct Primary Keys, 2 marks for 3 correct Primary Keys CUSTOMER: CustomerID ROOM: RoomNumber BOOKING: BookingID		
3(a)(iii)	1 mark for both table name and Foreign Key Table: BOOKING Foreign Key: CustomerID / RoomNumber		
3(b)	1 mark per bullet point to max 2 plus 1 mark for suitable example for each DBMS tool Developer Interface - To create user friendly features e.g. forms to enter new bookings - To create outputs e.g. report of bookings on a given date - To create interactive features e.g. buttons and menus Query processor - To create SQL/QBE queries - To search for data that meets set criteria, e.g. all bookings for next week - To perform calculations on extracted data, e.g. number of empty rooms tomorrow		
3c	1 mark for at least two correct rows, 2 marks for all four correct rows		
	Script	DDL	DML
	CREATE TABLE FILMS	✓	
	SELECT FilmID FROM FILMS		✓
	ALTER TABLE FILMS ADD PRIMARY KEY (FilmID)	✓	
	CREATE DATABASE MYDATA	✓	