

ANSWERS ASSEMBLY LANGUAGES

Answer 1

Question	Answer	Marks
6(a)(i)	1 mark for each correct answer A: The number 193 B: The data in memory location 193 C: The data in the memory location found by adding the contents of the IX to 193	3
6(a)(ii)	1 mark each correct answer <ul style="list-style-type: none"> • Indirect • Relative 	2

Question	Answer	Marks
6(b)	1 mark for correctly naming register, 1 mark for appropriate role <ul style="list-style-type: none"> • Program counter // PC • Stores the address of the next instruction to be fetched • Memory address register // MAR • Stores the address where data/instruction is to be read from or saved to • Memory data register // MDR • Stores data that is about to be written to memory // Stores data that has just been read from memory • Current instruction register // CIR • Stores the instruction that is currently being decoded/executed 	4

Answer 2

Question	Answer	Marks
5(a)	1 mark for one letter in the correct place, 2 marks for all three correct 2 B 4 A 5 C	2
5(b)(i)	11000010	1
5(b)(ii)	CD	1
5(b)(iii)	1 mark per bullet point to max 2 <ul style="list-style-type: none"> • The maximum range for an 8-bit two's complement binary number is -128 to +127 • ... 200 is outside of the maximum range 	2

5(c) 1 mark for each highlighted section block

Instruction address	ACC	Memory address			
		100	101	102	103
		1	2	3	0
20	0				
21					
22					
23	1				
24	3				
25		3			
26	0				
27	1				
28					1
29					
20	1				
21					
22					
23	3				
24	5				
25		5			
26	1				
27	2				
28					2
29					
20	2				
21					
22					
30					

Answer 3

Instruction address	ACC	Memory address							IX	OUTPUT
		100	101	102	103	104	300	301		
		65	67	69	69	68		33	0	
20	0									
21							0			
22										
23										
28	65									
29										A
30	0									
31	1									
32							1			
33									1	
34										
35										
22										
24	67									
25	100									
26										d
27										
30	1									
31	2									
32							2			
33									2	
34										
36										

Answer 4

Question	Answer	Marks																									
4(a)	1 mark per bullet <div><input type="checkbox"/> LDM #300 The (denary) number 300 is loaded (into the register)</div> <div><input type="checkbox"/> LDD 300 The <u>contents</u> of address 300 are loaded (into the register)</div>	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		✓																								

Answer 5

Question	Answer	Marks
2(a)(i)	1 mark for correct answer <ul style="list-style-type: none"> • To convert a (higher level) programming language to a different form 	1
2(a)(ii)	1 mark per bullet point to max 2 <ul style="list-style-type: none"> • Easier de-bugging • ...because errors can be corrected in real time • The effect of any change made to the code can be seen immediately • Parts of the program can be <u>tested</u>, without all the program code being available 	2
2(a)(iii)	1 mark for correct answer Assembler	1

Question	Answer	Marks
2(b)	1 mark per bullet point to max 2 <ul style="list-style-type: none"> • The code is already written so the programmer is not starting over again which saves time • The code will have been used by many people so it should be already thoroughly tested and relatively error-free / won't need re-testing • The programmer can use e.g. mathematical / graphics functions etc. that he may not know how to code • Use of named library functions can simplify the program and make it easier to read • The library routine code should conform to industry standards and therefore contribute towards a more robust program 	2

Answer 6

Instruction address	ACC	Memory address						IX	OUTPUT	
		300	301	302	303	400	401			
		2	5	0	4	64		0		
50	0									
51							0			[1]
52	2									[1]
53										
54										
55	66									[1]
56									B	[1]
57	0									[1]
58	1									
59							1			
60								1		[1]
61										[1]
52	5									
53										
54										
55	69									
56									E	
57	1									
58	2									[1]
59							2			
60								2		
61										
52	0									[1]
53										
54										
62										

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

Answer 7

Question	Answer	Marks												
7	1 mark for each correct addressing mode <table><tr><th>Addressing mode</th><th>Description</th></tr><tr><td>Relative</td><td>Form the address by adding the given number to a base address. Load the contents of the calculated address to the Accumulator (ACC).</td></tr><tr><td>Indirect</td><td>Load the contents of the address held at the given address to ACC.</td></tr><tr><td>Direct</td><td>Load the contents of the given address to ACC.</td></tr><tr><td>Indexed</td><td>Form the address from the given address + the contents of the Index Register. Load the contents of the calculated address to ACC.</td></tr><tr><td>Immediate</td><td>Load the given value directly to ACC.</td></tr></table>	Addressing mode	Description	Relative	Form the address by adding the given number to a base address. Load the contents of the calculated address to the Accumulator (ACC).	Indirect	Load the contents of the address held at the given address to ACC.	Direct	Load the contents of the given address to ACC.	Indexed	Form the address from the given address + the contents of the Index Register. Load the contents of the calculated address to ACC.	Immediate	Load the given value directly to ACC.	5
Addressing mode	Description													
Relative	Form the address by adding the given number to a base address. Load the contents of the calculated address to the Accumulator (ACC).													
Indirect	Load the contents of the address held at the given address to ACC.													
Direct	Load the contents of the given address to ACC.													
Indexed	Form the address from the given address + the contents of the Index Register. Load the contents of the calculated address to ACC.													
Immediate	Load the given value directly to ACC.													

Answer 8

Question	Answer	Marks												
4(a)(ii)	1 mark per bullet point Direct addressing: <input type="checkbox"/> 20 is the address of the data Indirect addressing: <input type="checkbox"/> 20 is an address which holds the address where the data is stored	2												
4(b)	1 mark for 1 correct tick 2 marks for 3 correct ticks <table border="1"> <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												

Instruction address	ACC	Memory address						IX (Index Register)	OUTPUT
		200	201	202	203	204	205		
70	130	130	133	130	0	0	2	0	
71									*
72					130				
73	0								
74	1								
75						1			
76								1	
77	133								
78									
79									
81	1								
82									
83									
74	2								
75						2			
76								2	
77	130								
78									
79									
80									*
81	2								
82									
83									
84									

Answer 9

3(a)(i)	1 mark per bullet point Absolute addressing: <ul style="list-style-type: none"> The operand is a numeric address // The numeric address is given // referring directly to a memory location Symbolic addressing: <ul style="list-style-type: none"> 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, ADD 230 Symbolic addressing: For example, ADD num1	2
3(b)(i)	1 mark per bullet point Indexed addressing: <ul style="list-style-type: none"> The address to be used is formed by: operand + the contents of the Index Register (IX) Immediate addressing: <ul style="list-style-type: none"> The operand is not an address // the operand is the actual value to be loaded 	2

Question	Answer	Marks
3(b)(ii)	1 mark per example Indexed: For example, LDX 20 Immediate: For example, ADD #20	2
3(c)(i)	193	1
3(c)(ii)	C1	1
3(c)(iii)	-63	1

Question		Answer							Marks
Instruction address	ACC	Memory address						IX	OUTPUT
		100	101	102	103	104	105		
		2	302	303	303	0	303	1	
40	2								
41									
43	302								
44									
45									
47	2								
48	1								
49		1							
50								2	
51									
41									
42									
43	303								
44									
45									
46									+
47	1								
48	0								
49		0							
50								3	
51									
41									
54									

Answer 10

Question	Answer	Marks
2(b)(ii)	F2	1
2(b)(iii)	−14	1
2(b)(iv)	<div> <div>1</div> <div>1</div> <div>1</div> <div>1</div> <div>0</div> <div>0</div> <div>1</div> <div>1</div> </div>	1
2(c)	1 mark per bullet point <ul style="list-style-type: none"> 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 32) Jumping, loading, incrementing, storing in 96, incrementing IX and end (instructions 24–32) 	7

Instruction address	ACC	Memory address						IX	OUTPUT
		93	94	95	96	97	98		
		453	453	452	8	10	453	8	
20	8								
21									
22									
23	453								
24									
25									
26									&
27	8								
28	9								
29					9				
30								9	
31									
21									
22									
23	452								
24									
25									
27	9								
28	10								
29					10				
30								10	
31									
21									
22									
32									

Answer 11

Question	Answer	Marks
4(a)(i)	500	1
4(a)(ii)	496	1
4(a)(iii)	502	1
4(a)(iv)	86	1
4(b)	<div> <div>00000001</div> <div>00010001</div> <div>00000110</div> <div>01100001</div> </div> <p>Both correct op codes Operand 0001 0001 Operand 0110 0001</p>	<div>1</div> <div>1</div> <div>1</div>
4(c)	256	1
4(d)(i)	07 C2 07 C2	<div>1</div> <div>1</div>
4(d)(ii)	LDI 63 LDI 63	<div>1</div> <div>1</div>

Answer 12

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

Question	Answer	Marks
5(a)(iv)	86	1
5(b)	<div> <div>Op code</div> <div>000100010</div> <div>00010101</div> <div>Operand</div> <div>01000011</div> <div>00000111</div> </div> <p>Both correct op codes Operand 0100 0011 Operand 0000 0111</p>	<div>1</div> <div>1</div> <div>1</div>
5(c)(i)	14 5E 14 5E	<div>1</div> <div>1</div>
5(c)(ii)	LDR #77 LDR #77	<div>1</div> <div>1</div>

Answer 13

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> <div style="display: flex; align-items: center; margin-left: 20px;"> <div style="margin-right: 10px;"> <p>(1)</p> <p>(1) + (1)</p> <p>(1)</p> <p>(1)</p> </div> <div style="font-size: 3em;">}</div> </div>	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)	<p>Three marks from:</p> <ul style="list-style-type: none"> <input type="checkbox"/> The assembler scans the assembly language instructions in sequence <input type="checkbox"/> When it meets a symbolic address checks to see if already in symbol table <input type="checkbox"/> If not, it adds it to the symbol table in the symbolic address column <input type="checkbox"/> If it is already in symbol table check if absolute address known <input type="checkbox"/> If the absolute address is known, it is entered in the appropriate cell <input type="checkbox"/> 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)	<p>A – 1110 0110 0110 1000</p> <p style="text-align: center;">(1) (1)</p> <p>B – E6 68 (1)</p>	3																														

Answer 14

(b)

Instruction address	Working space	ACC	Memory address				IX	OUTPUT
			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								

Answer 15

- 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]

Answer 16

- 4 (a) 11001110

[1]

- (b)

[7]

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								

Answer 17

8 (a) (i)

Accumulator:

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

[1]

(ii) **One mark** for answer and **two marks** for explanationAccumulator:

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]

Answer 18

(a) (i)

0	1	1	1	0	0	1	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]

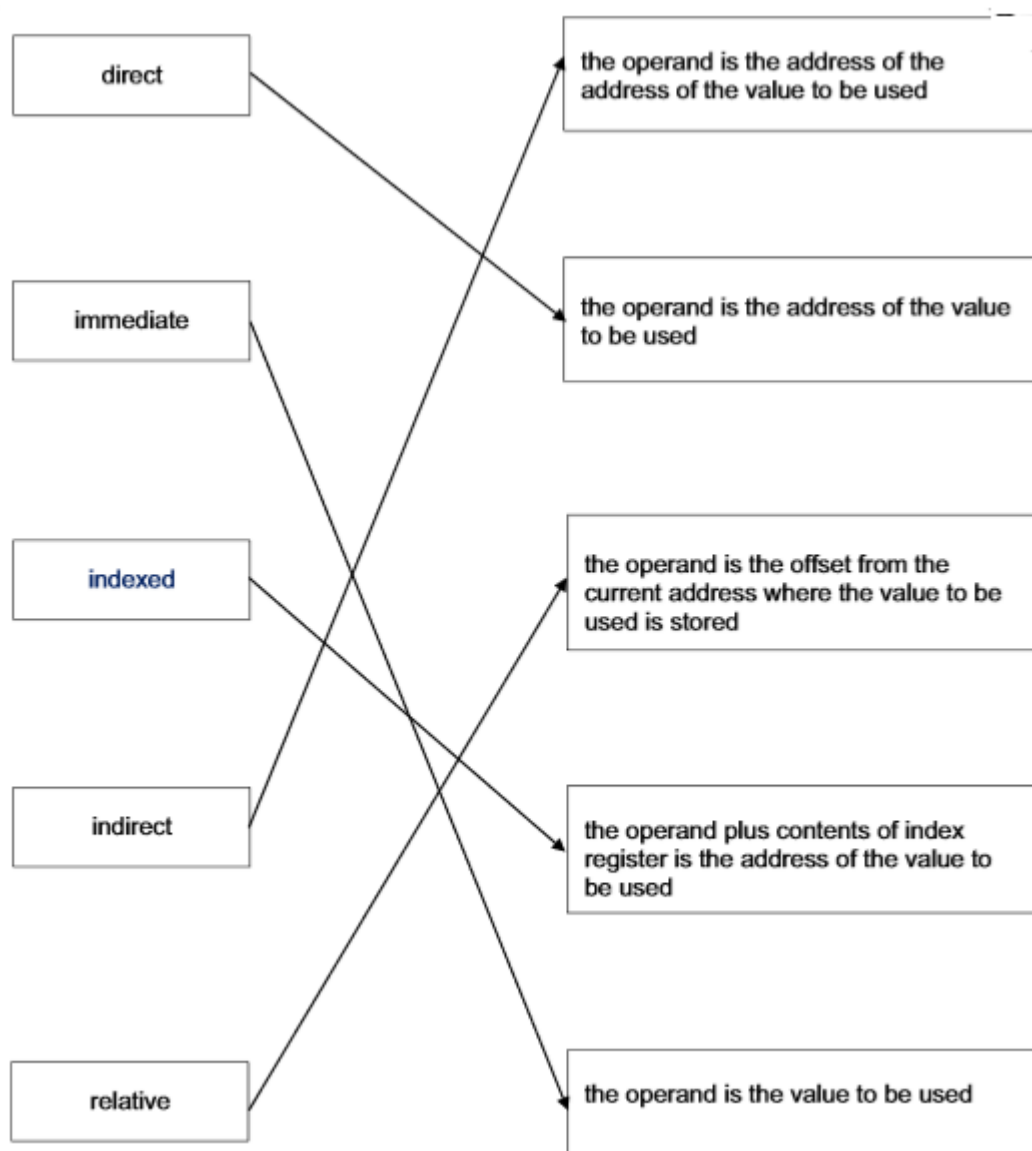
(b) **ONE** mark for each correct row.

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

[6]

Answer 19

3



[4]

Answer 20

(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]