4 The following table shows part of the instruction set for a processor. The processor has two registers: the Accumulator (ACC) and an Index Register (IX).

Instruction		Explanation
Opcode	**Operand**	
LDM	#n	Immediate addressing. Load the number n to ACC
LDD	<address>	Direct addressing. Load the contents of the location at the given address to ACC
LDI	<address>	Indirect addressing. The address to be used is at the given address. Load the contents of this second address to ACC
LDX	<address>	Indexed addressing. Form the address from <address> + the contents of the Index Register. Copy the contents of this calculated address to ACC
LDR	#n	Immediate addressing. Load the number n to IX
ADD	#n/Bn/&n	Add the number n to the ACC
ADD	<address>	Add the contents of the given address to the ACC
SUB	#n/Bn/&n	Subtract the number n from the ACC
SUB	<address>	Subtract the contents of the given address from the ACC
INC	<register>	Add 1 to the contents of the register (ACC or IX)

<address> can be an absolute or a symbolic address
# denotes a denary number, e.g. #123
B denotes a binary number, e.g. B01001010
& denotes a hexadecimal number, e.g. &4A

(a) The current contents of memory are shown:

Address	Data
19	24
20	2
21	1
22	3
23	5
24	4
25	22

The current contents of the ACC and IX are shown:

ACC	12
IX	1

Complete the table by writing the content of the ACC after each program has run.

Program number	Code	ACC content
1	LDD 20 ADD #2	
2	LDX 22	
3	LDI 25 INC ACC SUB 22	
4	LDD 19 LDM #5 LDM #25	

[4]

(b) The processor includes these bit manipulation instructions:

Instruction		Explanation
Opcode	**Operand**	
AND	#n/Bn/&n	Bitwise AND operation of the contents of ACC with the operand
AND	<address>	Bitwise AND operation of the contents of ACC with the contents of <address>
XOR	#n/Bn/&n	Bitwise XOR operation of the contents of ACC with the operand
XOR	<address>	Bitwise XOR operation of the contents of ACC with the contents of <address>
OR	#n/Bn/&n	Bitwise OR operation of the contents of ACC with the operand
OR	<address>	Bitwise OR operation of the contents of ACC with the contents of <address>

<address> can be an absolute or a symbolic address
# denotes a denary number, e.g. #123
B denotes a binary number, e.g. B01001010
& denotes a hexadecimal number, e.g. &4A

The current contents of memory are shown:

Address	Data
30	01110101
31	11111111
32	00000000
33	11001100
34	10101010

The current content of the ACC is shown:

1	0	0	1	1	0	1	0

Complete the table by writing the content of the ACC after each program has run.

The binary number 10011010 is reloaded into the ACC before each program is run.

Program number	Code	ACC content
1	AND 31	
2	XOR B01001111	
3	OR #30	

[3]