QUESTION 7.

9 A database has been designed to store data about salespersons and the product

The following facts help to define the structure of the database:

- each salesperson works in a particular shop
- each salesperson has a unique first name
- each shop has one or more salespersons
- each product which is sold is manufactured by one company only
- each salesperson can sell any of the products
- the number of products that each salesperson has sold is recorded

The table `ShopSales` was the first attempt at designing the database.

FirstName	Shop	ProductName	NoOfProducts	Manufacturer
Nick	TX	television set refrigerator digital camera	3 2 6	SKC WP HKC
Sean	BH	hair dryer electric shaver	1 8	WG BG
John	TX	television set mobile phone digital camera toaster	2 8 4 3	SKC ARC HKC GK

(a) State why the table is **not** in First Normal Form (1NF).

.....

.....[1]

Question 9 begins on page 14.

QUESTION 8.

11 A game program is written which can be either interpreted or compiled. The table below shows five statements about the use of interpreters and compilers.

Tick (✓) to show whether the statement refers to an interpreter or to a compiler.

Statement	Interpreter	Compiler
This translator creates an executable file		
When this translator encounters a syntax error, game execution halts		
The translator analyses and checks each line just before executing it		
This translator will produce faster execution of the game program		
Use of this translator makes it more difficult for the user to modify the code of the game		

[5]

(b) (i) A black and white image is 512 pixels by 256 pixels.

Calculate the file size of this image in kilobytes (KB) (1 KB = 1024 bytes).
Show your working.

.....

.....

.....

.....

.....[2]

(ii) Give a reason why it is important to estimate the file size of an image.

.....

.....

.....[1]

9 (a) Give a brief description of each of the following terms:

Validation .....

.....

.....

Verification .....

.....

.....[2]

(b) Data are to be transferred between two devices. Parity checks are carried out on the data.

Explain what is meant by a parity check. Give an example to illustrate your answer.

.....

.....

.....

.....

.....

.....

.....

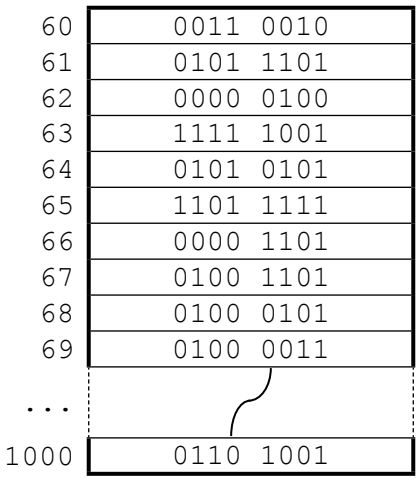
.....[4]

QUESTION 9.

9 The table shows assembly language instructions for a processor which has one general purpose register, the Accumulator (ACC) and an index register (IX).

Instruction		Explanation
Op code	**Operand**	
LDD	<address>	Direct addressing. Load the contents of the given 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.
STO	<address>	Store contents of ACC at the given address.
ADD	<address>	Add the contents of the given address to ACC.
INC	<register>	Add 1 to the contents of the register (ACC or IX).
DEC	<register>	Subtract 1 from the contents of the register (ACC or IX).
CMP	<address>	Compare contents of ACC with contents of <address>.
JPE	<address>	Following a compare instruction, jump to <address> if the compare was True.
JPN	<address>	Following a compare instruction, jump to <address> if the compare was False.
JMP	<address>	Jump to the given address.
OUT		Output to screen the character whose ASCII value is stored in ACC.
END		Return control to the operating system.

(a) The diagram shows the current contents of a section of main memory and the index register:



Index register:

0	0	0	0	1	0	0	0

Question 9 begins on page 12.

QUESTION 10.

6 **(a)** Explain the difference between the World Wide Web (WWW) and the Internet.

.....

.....

.....

.....[2]

(b) Three methods of connecting devices include fibre-optic cables, copper cables and radio waves. The table below gives descriptions relating to these connection methods.

Tick (✓) one box on each row to show the method that best fits each description.

Description	Fibre-optic cable	Copper cable	Radio waves
Wireless medium			
Twisted-pair is an example			
Uses light waves			
WiFi			
Fastest transmission medium			

[5]

(c) Bit streaming is used for both real-time and on-demand services.

Describe **one** difference between real-time and on-demand bit streaming.

.....

.....

.....

.....[2]

(d) A device needs an IP address to connect to the Internet. IPv4 is the more common type of IP address.

Describe, using an example, the format of an IPv4 address.

.....

.....

.....

.....

.....

.....[3]

(e) A computer user keys in the Uniform Resource Locator (URL) of a web p
browser.

Describe how the browser uses the Domain Name Service (DNS) to display the web

.....

.....

.....

.....

.....

.....

.....

.....[4]

QUESTION 11.

8 The table shows assembly language instructions for a processor which has one g
register, the Accumulator (ACC) and an Index Register (IX).

Instruction		Explanation
Op code	**Operand**	
LDD	<address>	Direct addressing. Load the contents of the given 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.
STO	<address>	Store contents of ACC at the given address.
ADD	<address>	Add the contents of the given address to ACC.
CMP	<address>	Compare contents of ACC with contents of <address>
JPE	<address>	Following a compare instruction, jump to <address> if the compare was True.
JPN	<address>	Following a compare instruction, jump to <address> if the compare was False.
JMP	<address>	Jump to the given address.
OUT		Output to the screen the character whose ASCII value is stored in ACC.
END		Return control to the operating system.

The diagram shows the contents of the main memory:

Main memory

800	0110 0100
801	0111 1100
802	1001 0111
803	0111 0011
804	1001 0000
805	0011 1111
806	0000 1110
807	1110 1000
808	1000 1110
809	1100 0010
:	
2000	1011 0101

(a) (i) Show the contents of the Accumulator after execution of the instruction:

```
LDD  802
```

Accumulator: | | | | | | | | |

[1]

(ii) Show the contents of the Accumulator after execution of the instruction:

```
LDX 800
```

Index Register:	0	0	0	0	1	0	0	1
Accumulator:								

Explain how you arrived at your answer.

.....

.....

.....

.....[3]

(b) (i) Complete the trace table below for the following assembly language program contains denary values.

100	LDD 800
101	ADD 801
102	STO 802
103	LDD 803
104	CMP 802
105	JPE 107
106	JPN 110
107	STO 802
108	OUT
109	JMP 112
110	LDD 801
111	OUT
112	END
⋮	
800	40
801	50
802	0
803	90

Selected values from the ASCII character set:

ASCII code	40	50	80	90	100
Character	(	2	P	Z	d

Trace table:

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

[4]

(ii) There is a redundant instruction in the code in **part (b)(i)**.

State the address of this instruction.

.....

(c) The program used the ASCII coding system for character codes. An alternative coding system is Unicode.

(i) Give **two** disadvantages of using ASCII code.

1 .....

.....

2 .....

.....[2]

(ii) Describe how Unicode is designed to overcome the disadvantages of ASCII.

.....

.....

.....

.....[2]

BLANK PAGE

BLANK PAGE

16

BLANK PAGE

QUESTION 12.

9 A health club offers classes to its members. A member needs to book into each c

(a) The health club employs a programmer to update the class booking system. The p has to decide how to store the records. The choice is between using a relational dat a file-based approach.

Give **three** reasons why the programmer should use a relational database.

1 .....

.....

.....

.....

2 .....

.....

.....

.....

3 .....

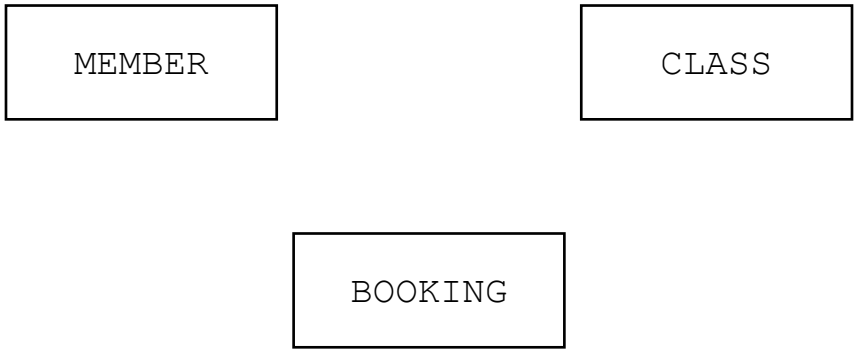
.....

.....

.....[6]

(b) The programmer decides to use three tables: `MEMBER`, `BOOKING` and `CLASS`.

Complete the Entity-Relationship (E-R) diagram to show the relationships between these tables.



[2]

QUESTION 13.

8 The table shows assembly language instructions for a processor which has one general register, the Accumulator (ACC) and an Index Register (IX).

Instruction		Explanation
Op code	**Operand**	
LDD	<address>	Direct addressing. Load the contents of the given 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.
STO	<address>	Store contents of ACC at the given address.
ADD	<address>	Add the contents of the given address to ACC.
CMP	<address>	Compare contents of ACC with contents of <address>
JPE	<address>	Following a compare instruction, jump to <address> if the compare was True.
JPN	<address>	Following a compare instruction, jump to <address> if the compare was False.
JMP	<address>	Jump to the given address.
OUT		Output to the screen the character whose ASCII value is stored in ACC.
END		Return control to the operating system.

The diagram shows the contents of the main memory:

Main memory

800	0110 0100
801	0111 1100
802	1001 0111
803	0111 0011
804	1001 0000
805	0011 1111
806	0000 1110
807	1110 1000
808	1000 1110
809	1100 0010
:	
2000	1011 0101

(a) (i) Show the contents of the Accumulator after execution of the instruction:

```
LDD  802
```

Accumulator: | | | | | | | | |

[1]

(ii) Show the contents of the Accumulator after execution of the instruction:

```
LDX 800
```

Index Register:	0	0	0	0	1	0	0	1
Accumulator:								

Explain how you arrived at your answer.

.....

.....

.....

.....[3]

(b) (i) Complete the trace table below for the following assembly language program contains denary values.

100	LDD 800
101	ADD 801
102	STO 802
103	LDD 803
104	CMP 802
105	JPE 107
106	JPN 110
107	STO 802
108	OUT
109	JMP 112
110	LDD 801
111	OUT
112	END
⋮	
800	40
801	50
802	0
803	90

Selected values from the ASCII character set:

ASCII code	40	50	80	90	100
Character	(	2	P	Z	d

Trace table:

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

[4]

(ii) There is a redundant instruction in the code in **part (b)(i)**.

State the address of this instruction.

.....

(c) The program used the ASCII coding system for character codes. An alternative coding system is Unicode.

(i) Give **two** disadvantages of using ASCII code.

1 .....

.....

2 .....

.....[2]

(ii) Describe how Unicode is designed to overcome the disadvantages of ASCII.

.....

.....

.....

.....[2]

BLANK PAGE

BLANK PAGE

16

BLANK PAGE

QUESTION 14.

5 A computer receives data from a remote data logger. Each data block is a group
A block is made up of seven data bytes and a parity byte.

Each data byte has a parity bit using odd parity. The parity byte also uses odd parity.

The following table shows a data block before transmission. Bit position 0 is the parity bit.

Bit position							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
1	1	0	0	1	1	0	1
0	0	1	0	0	0	0	0
1	0	0	1	1	1	0	**A**
1	1	0	0	0	0	1	0
1	1	0	0	0	0	1	0
1	1	0	0	0	1	1	**B**
0	0	0	0	0	0	0	0

Data bytes

0	1	1	0	1	1	0	0

← Parity byte

(a) (i) Describe how the data logger calculates the parity bit for each of the bytes in the data block.

.....

.....

.....[2]

(ii) State the two missing parity bits labelled **A** and **B**.

A = ...

B = ... [1]

(iii) Describe how the computer uses the parity byte to perform a further check on the received data bytes.

.....

.....

.....[2]

(b) (i) A second data block is received as shown in the following table. There
data block.

Identify and then circle **two** bits in the table which must be changed to remove th

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]

(ii) Explain how you arrived at your answers for **part (b)(i)**.

.....

.....

.....

.....[3]

QUESTION 15.

5 The following table shows part of the instruction set for a processor. The pro
general purpose register, the Accumulator (ACC), and an Index Register (IX).

Instruction			Explanation
Op code (mnemonic)	**Operand**	**Op code (binary)**	
LDD	<address>	0001 0011	Direct addressing. Load the contents of the location at the given address to the Accumulator (ACC).
LDI	<address>	0001 0100	Indirect addressing. The address to be used is at the given address. Load the contents of this second address to ACC.
LDX	<address>	0001 0101	Indexed addressing. Form the address from <address> + the contents of the Index Register. Copy the contents of this calculated address to ACC.
LDM	#n	0001 0010	Immediate addressing. Load the denary number n to ACC.
LDR	#n	0001 0110	Immediate addressing. Load denary number n to the Index Register (IX).
STO	<address>	0000 0111	Store the contents of ACC at the given address.

The following diagram shows the contents of a section of main memory and the Index Register (IX).

(a) Show the contents of the Accumulator (ACC) after each instruction is executed.

IX	0	0	0	0	0	1	1	0

(i) LDD 355

ACC ..... [1]

(ii) LDM #355

ACC ..... [1]

(iii) LDX 351

ACC ..... [1]

(iv) LDI 355

ACC ..... [1]

Address	Main memory contents
350	
351	86
352	
353	
354	
355	351
356	
357	22
358	

(b) Each machine code instruction is encoded as 16 bits (8-bit op code follo
operand).

Write the machine code for these instructions:

```
LDM #67
```


```
LDX #7
```


[3]

(c) Computer scientists often write binary representations in hexadecimal.

(i) Write the hexadecimal representation for the following instruction.

0	0	0	1	0	1	0	0	0	1	0	1	1	1	1	0

.....[2]

(ii) A second instruction has been written in hexadecimal as:

```
16 4D
```

Write the assembly language for this instruction with the operand in denary.

.....[2]

QUESTION 16.

7 A clinic is staffed by several doctors. The clinic serves thousands of patients. Eac
one time, there is only one doctor in the clinic available for appointments.

The clinic stores patient, doctor and appointment data in a relational database.

(a) (i) Underline the primary key for each table in the following suggested table designs.

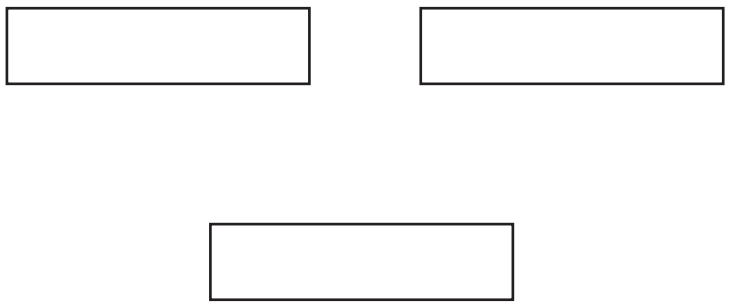
```
PATIENT(PatientID, PatientName, Address, Gender)

DOCTOR(DoctorID, Gender, Qualification)

APPOINTMENT(AppointmentDate, AppointmentTime, DoctorID, PatientID)
```

[2]

(ii) Complete the following entity-relationship (E-R) diagram for this design.



[2]

(b) The doctors are concerned that many patients make appointments but do not attend them.

Describe the changes to the table designs that could be made to store this information.

.....

.....[2]

Issue 2

Description .....

.....

.....

ACM/IEEE principle (Circle one only)

Public	Client and Employer	Product	Judgement
Management	Profession	Colleagues	Self

Possible action .....

.....

.....

[6]

Question 7 begins on the next page.

QUESTION 17.

7 A company takes customer service for its clients very seriously.

The client

- The client names are unique.

A visit

- The company arranges a date for a visit to gather feedback from a client.
- A visit to a client never takes more than one day.
- Over time, the client receives many visits.

Staff (Interviewers)

- One or more staff attend the visit.
- If there is more than one staff member visiting, each performs a separate interview.

Interviews

- Each interview is classified as either 'general' or by some specialism, for example, marketing, customer service or sales.
- A report is produced for each interview, `InterviewText`.
- Each interview is conducted by a single staff member.

The client, visit, staff and interview data will be stored in a relational database.

(a) **(i)** Underline the primary key for each table in the following suggested table designs.

```
STAFF(StaffID, StaffName, Department)

CLIENT(ClientName, Address, Town)

VISIT(ClientName, VisitDate)

INTERVIEW(ClientName, VisitDate, StaffID, SpecialistFocus, InterviewText)
```

[3]

(ii) For each of the pairs of entities, A, B and C, draw the relationship between the two entities.

A | CLIENT | VISIT |

B | VISIT | INTERVIEW |

C | INTERVIEW | STAFF |

[3]

6 **(a)** A personal computer (PC) is extensively used for a wide range of applications, including the three shown in the following table.

Write in the table, a suitable input device, output device, or both needed for each application.

Do not give a monitor, keyboard or mouse in your answers.

Application	Input device	Output device
Capture the text from a paper document, in order that the text can be word processed		
Producing a replica of a small plastic component from a washing machine		
A museum has interactive information facilities throughout the building		

[3]

(b) Explain the basic internal operation of a hard disk drive.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....[4]

QUESTION 18.

7 A movie theatre has a relational database that stores the movie schedule, and i
the movies. The theatre has several screens that play movies at the same time.

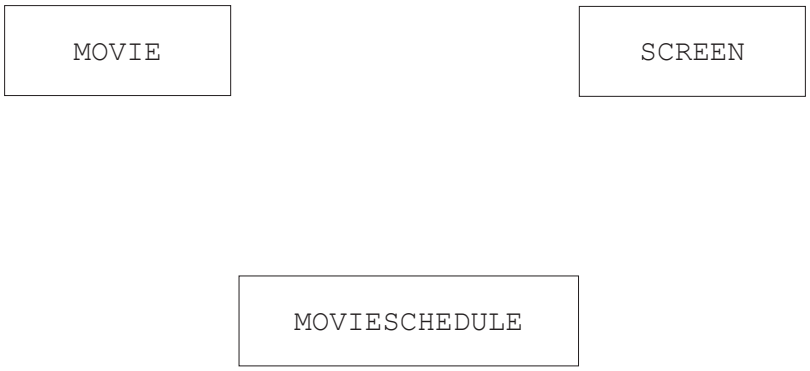
The database has three tables to store information about the movies, the screens and t
schedule.

MOVIE(MovieID, Title, Length, Rating)

SCREEN(ScreenNumber, NumberSeats)

MOVIESCHEDULE(ScheduleID, MovieID, ScreenNumber, Time)

(a) Complete the entity-relationship (E-R) diagram to show the relationships between these tables.



[2]

(b) Explain how primary and foreign keys are used to link the tables in the movie theatre database.

.....

.....

.....

.....

.....

.....

.....[4]

(c) The database needs to store the name of the company that produced example, `Rocking Movies`.

Write an SQL script to add the attribute `ProductionCompany` to the `MOVIE` table.

.....

.....

.....

.....

.....[2]

(d) Write an SQL script to display the title and rating of all movies scheduled to play on screen number 3.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....[4]

QUESTION 19.

3 The following table shows assembly language instructions for a processor which purpose register, the Accumulator (ACC) and an Index Register (IX).

Instruction		Explanation
Op code	**Operand**	
LDD	<address>	Direct addressing. Load the contents of the location at the given 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.
STO	<address>	Store contents of ACC at the given address.
ADD	<address>	Add the contents of the given address to ACC.
INC	<register>	Add 1 to the contents of the register (ACC or IX).
DEC	<register>	Subtract 1 from the contents of the register (ACC or IX).
CMP	<address>	Compare contents of ACC with contents of <address>.
JPE	<address>	Following compare instruction, jump to <address> if the compare was True.
JPN	<address>	Following compare instruction, jump to <address> if the compare was False.
JMP	<address>	Jump to the given address.
OUT		Output to the screen the character whose ASCII value is stored in ACC.
END		Return control to the operating system.

(a) **(i)** State what is meant by **absolute addressing** and **symbolic addressing**.

Absolute addressing .....

.....

Symbolic addressing .....

.....

[2]

(ii) Give an example of an ADD instruction using both absolute addressing and symbolic addressing.

Absolute addressing .....

Symbolic addressing .....

[2]

(b) (i) State what is meant by **indexed addressing** and **immediate addressi**

Indexed addressing .....

.....

Immediate addressing .....

.....

[2]

(ii) Give an example of an instruction that uses:

Indexed addressing .....

Immediate addressing .....

[2]

(c) The current contents of a general purpose register (X) are:

X	1	1	0	0	0	0	0	1

(i) The contents of X represent an unsigned binary integer.

Convert the value in X into denary.

..... [1]

(ii) The contents of X represent an unsigned binary integer.

Convert the value in X into hexadecimal.

..... [1]

(iii) The contents of X represent a two's complement binary integer.

Convert the value in X into denary.

..... [1]

(d) The current contents of the main memory, Index Register (IX) and selected ASCII character set are:

Address	Instruction
40	LDD 100
41	CMP 104
42	JPE 54
43	LDX 100
44	CMP 105
45	JPN 47
46	OUT
47	LDD 100
48	DEC ACC
49	STO 100
50	INC IX
51	JMP 41
52	
53	
54	END
…	
100	2
101	302
102	303
103	303
104	0
105	303

IX	1

ASCII code table (selected codes only)

ASCII code	Character
300	/
301	*
302	-
303	+
304	^
305	=

This is a copy of the instruction set.

Instruction		Explanation
Op code	**Operand**	
LDD	<address>	Direct addressing. Load the contents of the location at the given 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.
STO	<address>	Store contents of ACC at the given address.
ADD	<address>	Add the contents of the given address to ACC.
INC	<register>	Add 1 to the contents of the register (ACC or IX).
DEC	<register>	Subtract 1 from the contents of the register (ACC or IX).
CMP	<address>	Compare contents of ACC with contents of <address>.
JPE	<address>	Following a compare instruction, jump to <address> if the compare was True.
JPN	<address>	Following a compare instruction, jump to <address> if the compare was False.
JMP	<address>	Jump to the given address.
OUT		Output to the screen the character whose ASCII value is stored in ACC.
END		Return control to the operating system.

9

Complete the trace table for the given assembly language program.

Instruction address	ACC	Memory address						IX	Ou
		100	101	102	103	104	105		
		2	302	303	303	0	303	1	
40									

[7]

QUESTION 20.

5 A simple program written in assembly language is translated using a two-pass as

(a) The table contains some of the tasks performed by a two-pass assembler.

Tick (✓) **one** box in each row to indicate whether the task is performed at the first or s pass. The first row has been completed for you.

Task	First pass	Second pass
Creation of symbol table	✓	
Expansion of macros		
Generation of object code		
Removal of comments		

[2]

(b) The processor's instruction set can be grouped according to their function. For example, one group is modes of addressing.

Identify **two** other groups of instructions.

1 .....

.....

2 .....

.....

[2]

(c) The table shows assembly language instructions for a processor which h
purpose register, the Accumulator (ACC), and an Index Register (IX).

Instruction		Explanation
Op code	**Operand**	
LDM	#n	Immediate addressing. Load the denary number n to ACC.
LDD	<address>	Direct addressing. Load the contents of the location at the given address t 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 denary number n to IX.
STO	<address>	Store contents of ACC at the given address.
ADD	<address>	Add the contents of the given address to ACC.
INC	<register>	Add 1 to the contents of the register (ACC or IX).
CMP	#n	Compare contents of ACC with denary number n.
JPE	<address>	Following a compare instruction, jump to <address> if the compare was True.
JPN	<address>	Following a compare instruction, jump to <address> if the compare was False.
JMP	<address>	Jump to the given address.
OUT		Output to screen the character whose ASCII value is stored in ACC.
END		Return control to the operating system.

The current contents of the main memory, Index Register (IX) and selected values from the ASCII character set are:

Address	Instruction
20	LDM #0
21	STO 300
22	CMP #0
23	JPE 28
24	LDX 100
25	ADD 301
26	OUT
27	JMP 30
28	LDX 100
29	OUT
30	LDD 300
31	INC ACC
32	STO 300
33	INC IX
34	CMP #2
35	JPN 22
36	END
...	
100	65
101	67
102	69
103	69
104	68
...	
300	
301	33
IX	0

ASCII code table (Selected codes only)

ASCII Code	Character
65	A
66	B
67	C
68	D
69	E
97	a
98	b
99	c
100	d
101	e

Trace the program currently in memory using the following trace table. The
has been completed for you.

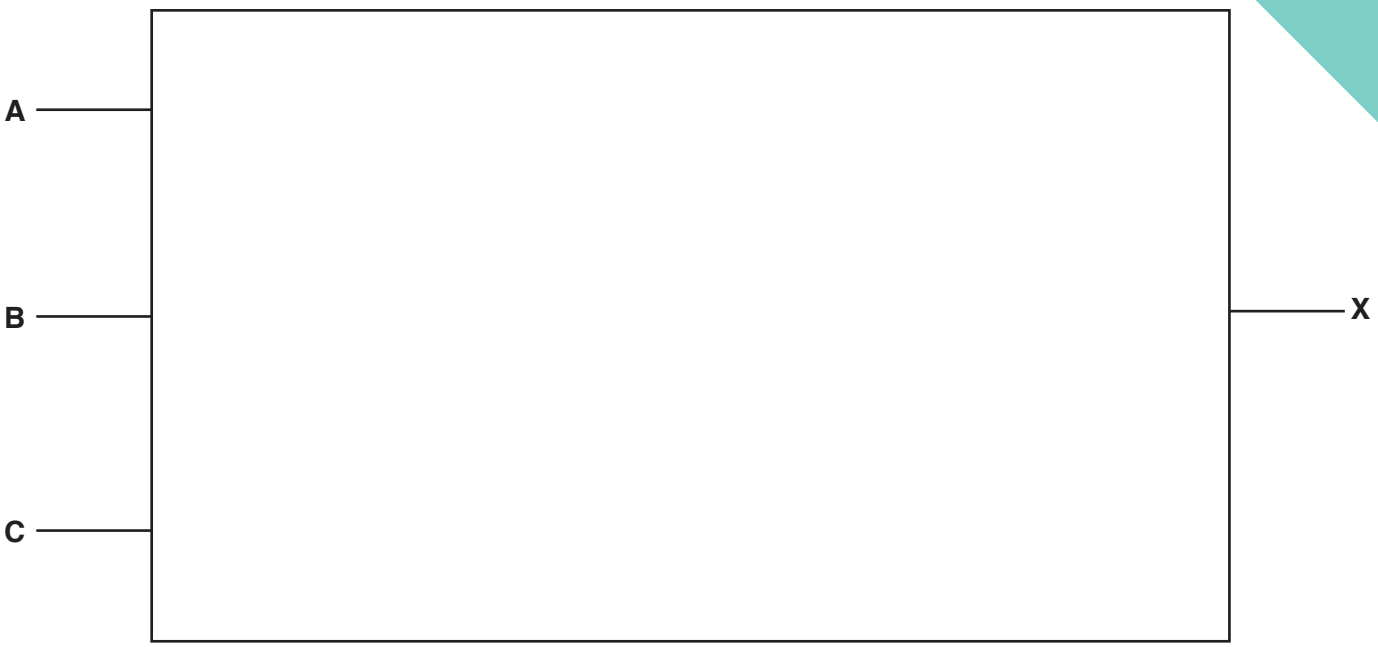
Instruction address	ACC	Memory address							IX	OU
		100	101	102	103	104	300	301		
		65	67	69	69	68		33	0	
20	0									

[8]

QUESTION 21.

6 **(a)** Draw a logic circuit to represent the logic expression:

X = **A** OR (**B** AND NOT **C**) OR (**A** AND **B**)



[5]

(b) Complete the truth table for the logic expression in **part (a)**.

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]

BLANK PAGE

BLANK PAGE

QUESTION 22.

4 A program is written in assembly language.

(a) The op codes `LDM` and `LDD` are used to load a register. The op code `LDM` uses addressing, and the op code `LDD` uses direct addressing.

Describe what happens when the following instructions are run.

```
LDM #300
```

.....

.....

```
LDD 300
```

.....

.....

[2]

(b) Assembly language instructions can be grouped by their purpose.

The following table shows four assembly language instructions.

Tick (✓) **one** box in each row to indicate the group each instruction belongs to.

Instruction	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]

(c) The processor handles interrupts within the fetch-execute cycle.

(i) Give **one** example of a hardware interrupt and **one** example of a software int

Hardware .....

.....

Software .....

.....

[2]

(ii) Explain how the processor handles an interrupt.

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... [5]

QUESTION 23.

4 Customers of a bank can access their account information by logging in on the ba

(a) The bank has a client-server model of networked computers.

(i) Describe, using the bank as an example, the key features of a client-server model.

.....

.....

.....

.....

.....

..... [3]

(ii) Give **two** other examples of applications that can use the client-server model.

1 .....

.....

2 .....

.....

[2]

(b) The bank's customers log in to the website using a web application.

Explain why the web application uses server-side scripting.

.....

.....

.....

.....

.....

..... [3]

(c) The bank is upgrading its local area network (LAN) copper cables to fibre-op

(i) State **two** benefits to the bank of upgrading to fibre-optic cable from copper c

1 .....

.....

2 .....

.....

[2]

(ii) State **two** drawbacks of upgrading to fibre-optic cables.

1 .....

.....

2 .....

.....

[2]

(d) The bank uses a relational database, ACCOUNTS, to store the information a
and their accounts.

The database stores the customer's first name, last name and date of birth.

The bank has several different types of account. Each account type has a unique ID numb
name (for example, regular or saving) and bonus (for example, $5.00, $10.00 or $15.00).

A customer can have more than one account.

Each customer's account has its own ID number and stores the amount of money the customer has in that account.

The bank creates a normalised, relational database to store the required information. There are three tables:

- CUSTOMER
- ACCOUNT_TYPE
- CUSTOMER_ACCOUNT

(i) Write the attributes for each table to complete the database design for the bank.

CUSTOMER(.....

.....

.....)

ACCOUNT_TYPE(.....

.....

.....)

CUSTOMER_ACCOUNT(.....

.....

.....)

[3]

(ii) Identify the primary key for each table that you designed in **part (d)(i)**.

CUSTOMER .....

ACCOUNT_TYPE .....

CUSTOMER_ACCOUNT .....

[2]

(iii) Identify **one** foreign key in one of the tables that you designed in **part (d)(i)**.

Table name .....

Foreign key .....

[1]

(iv) The following table has definitions of database terms.

Write the correct database term in the table for each definition.

Definition	Term
All the data about one entity	
The data in one row of a table	
A column or field in a table	

[3]