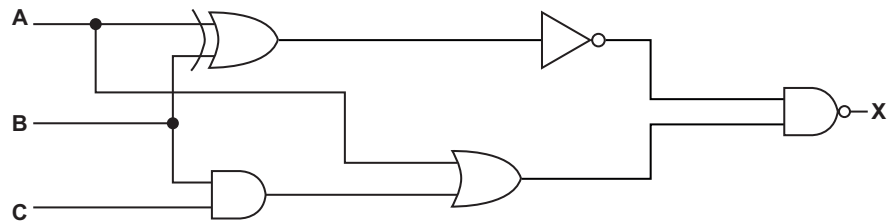


9 (a) Consider the following logic circuit:



Write the logic expression for the logic circuit. Do **not** simplify the expression.

X = [2]

(b) Consider the following logic expression:

$$X = \text{NOT } ((A \text{ NAND } B) \text{ XOR } (\text{NOT } A \text{ OR } \text{NOT } C))$$

A truth table for the logic expression is given:

Row number	A	B	C	X
1	0	0	0	1
2	0	0	1	0
3	0	1	0	1
4	0	1	1	0
5	1	0	0	1
6	1	0	1	0
7	1	1	0	1
8	1	1	1	1

There are **three** errors in the truth table.

Identify the **three** errors in the truth table by writing the row numbers with an incorrect output.

Error 1 Row number

Error 2 Row number

Error 3 Row number

[2]

11 The table shows assembly language instructions for a processor that has one register, the Accumulator (ACC).

Label	Instruction		Explanation
	Opcode	Operand	
	LDM	#n	Load the number n to ACC
	LDD	<address>	Load the contents of the location at the given address to the ACC
	LDI	<address>	The address to be used is at the given address. Load the contents of this second address to the ACC
	ADD	<address>	Add the contents of the given address to the ACC
	SUB	<address>	Subtract the contents of the given address from the ACC
	STO	<address>	Store the contents of the ACC at the given address
<label>:		<data>	Gives a symbolic address <label> to the memory location with contents <data>
# denotes a denary number, e.g. #123 <label> can be used in place of <address>			

(a) Write **assembly language** code, using **only** the given instruction set to:

- store the denary value 100 as a named constant
- subtract the constant from the value contained in address 632
- store the result in variable Answer.

Show the initialisation of the constant and Answer in the table provided.

.....

.....

.....

.....

.....

.....

Label	Contents

[6]

(b) The address 632 contains the value 45.

State the value of Answer after the code described in part (a) has executed.

1 A computer game is designed for users to select characters. Each character can take part in a group of events. Each group has five events. There are four types of event: jump, swim, run, drive.

The program is written using object-oriented programming.

(a) The class `EventItem` stores data about the events.

EventItem	
EventName : STRING	stores the name of the event
Type : STRING	stores the type of event, either: jump, swim, run or drive
Difficulty : INTEGER	stores the difficulty of the event from 1 (easiest) to 5 (hardest)
Constructor()	initialises EventName, Type and Difficulty to its parameter values
GetName()	returns the name of the event
GetDifficulty()	returns the difficulty of the event
GetEventType()	returns the type of event

(i) Write program code to declare the class `EventItem` and its constructor.

Do **not** declare the other methods.

Use your programming language's appropriate constructor.

All attributes must be private.

If you are writing in Python, include attribute declarations, using comments.

Save your program as **Question1_N24**.

Copy and paste the program code into part **1(a)(i)** in the evidence document.

[4]

(ii) The get methods `GetName()`, `GetDifficulty()` and `GetEventType()` each return the relevant attribute.

Write program code for the **three** get methods.

Save your program.

Copy and paste the program code into part **1(a)(ii)** in the evidence document.

[3]

(b) The array `Group` stores five objects of type `EventItem`.

(i) Write program code to declare `Group` local to the main program.

Save your program.

Copy and paste the program code into part **1(b)(i)** in the evidence document.

[1]

(ii) One group has the following events:

Event name	Event type	Event difficulty
Bridge	jump	3
Water wade	swim	4
100 mile run	run	5
Gridlock	drive	2
Wall on wall	jump	4

Write program code to create an instance of `EventItem` for each of the **five** events, and store them in `Group`.

Save your program.

Copy and paste the program code into part **1(b)(ii)** in the evidence document.

[3]

(c) The class `Character` stores data about the characters in the game.

Each character has a skill level for each type of event. The skill level is an integer between 1 and 5 inclusive. Skill level 1 is the lowest skill level, and skill level 5 is the highest skill level.

Character	
CharacterName : STRING	stores the name of the character
Jump : INTEGER	stores the character's skill level at events of type jump
Swim : INTEGER	stores the character's skill level at events of type swim
Run : INTEGER	stores the character's skill level at events of type run
Drive : INTEGER	stores the character's skill level at events of type drive
Constructor()	initialises CharacterName, Jump, Swim, Run and Drive to its parameter values
GetName()	returns the name of the character
CalculateScore()	takes the type of event and difficulty as parameters. Calculates and returns the chance of the character completing the event

Write program code to declare the class `Character`, its constructor and get method.

Use your programming language's appropriate constructor.

All attributes must be private.

If you are writing in Python, include attribute declarations, using comments.

Save your program.

Copy and paste the program code into part 1(c) in the evidence document.

[4]

(d) The method `CalculateScore()` in the `Character` class calculates and returns the percentage chance of a character completing an event.

When a character's skill level is greater than or equal to the difficulty of that event, the percentage chance of completing the event is 100%.

When a character's skill level is less than the difficulty of that event, the character's skill level is subtracted from the difficulty of that event. This difference is used to identify the percentage chance of success using this table:

Difference	Percentage chance of success
1	80
2	60
3	40
4	20

For example:

- A character has a skill level of 3 for events of type run.
- An event of type run has a difficulty level of 5
- The character's skill level is less than the difficulty, therefore the difference is calculated.
- The difference is the character's skill level subtracted from the event difficulty, $5 - 3 = 2$
- The difference is 2, therefore the percentage chance of succeeding is 60%

Write program code for the method `CalculateScore()` to:

- take the type of event and difficulty as parameters
- calculate the percentage chance of the character completing the event
- return the percentage chance of completing the event as an integer number, for example 60

Save your program.

Copy and paste the program code into part 1(d) in the evidence document.

[4]

(e) Two characters are attempting each event in the group you created in part 1(b).

One character has the name Tarz and the skill levels:

Jump 5

Swim 3

Run 5

Drive 1

The second character has the name Geni and the skill levels:

Jump 2

Swim 2

Run 3

Drive 4

Each `Character` object is stored in a variable.

(i) Amend the main program to declare and create an instance of a `Character` object for Tarz and for Geni.

Save your program.

Copy and paste the program code into part 1(e)(i) in the evidence document.

[2]

(ii) Both characters Tarz and Geni take part in each event in the group you created in part 1(b).

These steps are repeated for all **five** events:

- The percentage chance of each character completing the event is calculated and compared.
- The character with the highest percentage chance of completing the event gets 1 point. Their character name is output together with a message telling them they have won that event.
- If both characters have the same percentage chance of completing the event, the scores do **not** change, and a message is output stating that the event is a draw.

When the total score for each character has been calculated, the name of the character with the highest score is output stating they have won and the number of points they have. If both characters have the same number of points, a message is output telling them the group is a draw.

Amend the main program to:

- calculate the score for each character in each event
- output the name of the character that wins each event or that the event is a draw
- output the name of the character that has the most points for the group or that the group is a draw.

Save your program.

Copy and paste the program code into part 1(e)(ii) in the evidence document.

[7]

(iii) Test your program.

Take a screenshot of the output.

Save your program.

Copy and paste the screenshot into part 1(e)(iii) in the evidence document.

[1]

3 A company sells online Computer Science courses to students in different countries.

The courses are stored on a public cloud.

(a) (i) Explain why the company uses a public cloud to store these courses.

.....

.....

.....

..... [2]

(ii) Describe **two** disadvantages of storing data on a public cloud compared to storing data on a server in a Local Area Network (LAN).

1

.....

.....

.....

2

.....

.....

..... [4]

(iii) State how the following security measures can be used to protect computer systems.

Firewall

.....

Encryption

.....

.....

Passwords

..... [3]

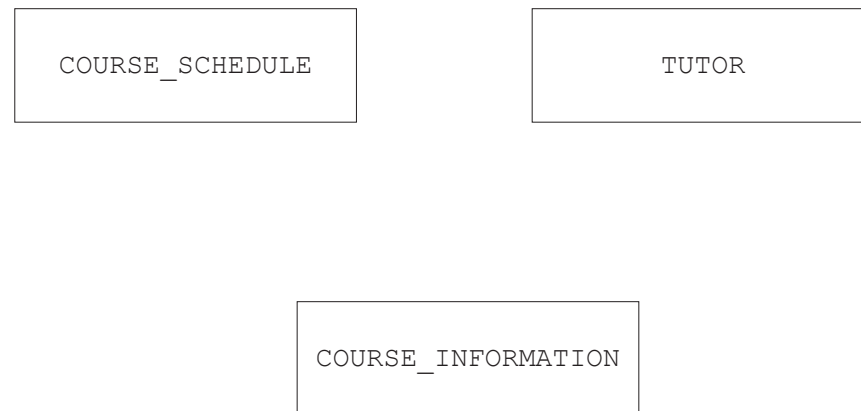
(b) The company uses a database, **COURSES**, to store data about the courses and their tutors.

Each course starts at different times of the year and may have a different tutor.

The database has the following structure:

```
COURSE_INFORMATION (CourseID, Description, Cost)
TUTOR (TutorID, TelephoneNumber, EmailAddress, TutorName)
COURSE_SCHEDULE (CourseID, DateStarted, TutorID)
```

(i) Complete the entity-relationship (E-R) diagram for the database **COURSES**.



[1]

(ii) Write the Structured Query Language (SQL) script to return the total number of courses that have started after 9 September 2023.

The value returned must have an appropriate field name.

.....

.....

.....

.....

.....

..... [4]

- (c) An example of a tutor ID is NK16C6.

An administrative officer enters the tutor ID into the TUTOR table.

Explain how data verification can be used when the tutor ID is entered.

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... [4]

- 3 A computer has an Operating System (OS).

- (a) Describe how the Operating System manages the peripheral hardware devices of the computer.

.....

.....

.....

.....

.....

.....

.....

.....

..... [4]

- (b) Hardware management is one key management task carried out by the Operating System.

Identify **two other** key management tasks carried out by the Operating System.

1

2 [2]

- (c) The Operating System has utility software including defragmentation software.

Explain how defragmentation can improve the performance of the computer.

.....

.....

.....

.....

.....

.....

..... [3]

- (d) The computer stores data in binary form.

- (i) State the difference between a kibibyte and a kilobyte.

.....

..... [1]

(ii) Convert the denary number 964 into Binary Coded Decimal (BCD).

 [1]

(iii) Convert the positive binary integer 11110010 into hexadecimal.

 [1]

(iv) Give the smallest **and** largest two's complement binary number that can be represented using 8 bits.
 Smallest
 Largest [2]

(v) Add the following two binary integers using binary addition. Show your working.

$$\begin{array}{r} 10110000 \\ + 00011011 \\ \hline \end{array}$$

[2]

(vi) Show the result of a 3-place right logical shift on the binary number:
 11001100
 [1]

5 A programmer is developing a computer game in a high-level language to sell to the public.

(a) The programmer uses both an interpreter and a compiler at different stages of the development of the program.

(i) Explain the reasons why the programmer uses an interpreter while writing the program code.

.....

 [2]

(ii) Explain the reasons why the programmer uses a compiler when the program has been written.

.....

 [3]

(b) The programmer needs to publish the game under a software licence so that it can be sold to the public.

Identify the **most appropriate** type of software licence for the game **and** justify your choice.

Licence

Justification

.....

 [4]