L-Università ta' Malta

MATSEC Examinations Board

Marking Scheme

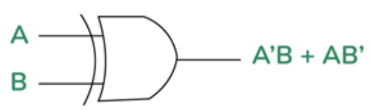
IM Computing

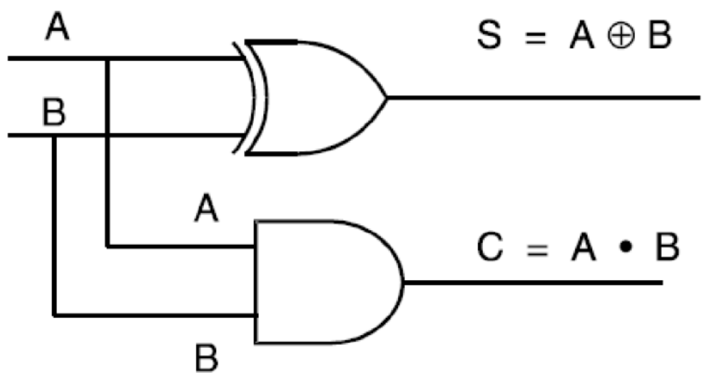
First Session 2025

Marking schemes published by the MATSEC Examination Board are not intended to be standalone documents. They are an essential resource for markers who are subsequently monitored through a verification process to ensure consistent and accurate application of the marking scheme.

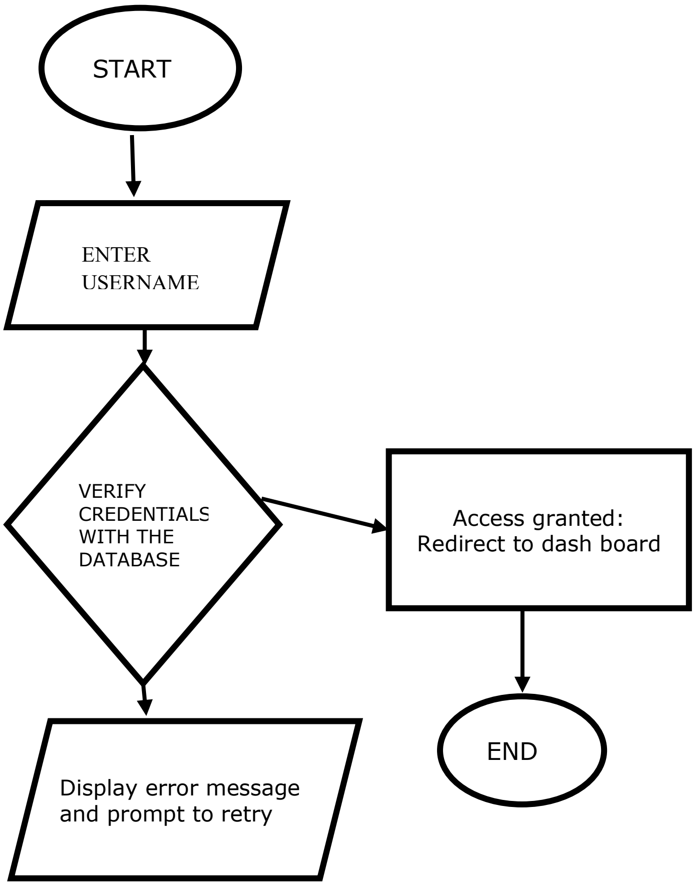
In the case of marking schemes that include model solutions or answers, it should be noted that these are not intended to be exhaustive. Variations and alternatives may also be acceptable. Examiners must consider all answers on their merits and will have consulted with the MATSEC Examinations Board when in doubt.

SECTION 'A'

Question No.			Solution	Mark Distribution	Marks
1	a		7.375 = 0111.0110	1 mark for correct integer part. 1 mark for correct fraction part	2
	b		1110.0100 = 14.25 or 14 ¼	1 mark for correct integer part. 1 mark for correct fraction part	2
	c		-17 = 11101111	1 mark for correct positive part. 1 mark for correct answer	2
Total:					**6**
2	a		$A\overline{A} + A\overline{B} + B\overline{A} + B\overline{B}$ Distributive law $0 + A\overline{B} + B\overline{A} + 0$ law of tautology AA' =0 $A\overline{B} + B\overline{A}$	1 mark for each law	2
	b		XOR Gate  		1

	c		 ii. Half Adder	1 mark for each gate	2 1
Total:					**6**
3	a		Eric Daphne Colin Beatrice Anthony Eric, Daphne, Colin, Beatrice, Anthony Pop – last in first out	1 mark for correct stack 1 mark for correct output	2
	b	i	Linear search is a simple search algorithm used to find the position of a target value within a list or array. It checks each element in the list one by one until the desired value is found or the end of the list is reached.		1

		ii	Example: Consider the array: [5, 8, 3, 10, 15] Criteria = 10: Start at index 0: Check 5. It's not 10. Move to index 1: Check 8. It's not 10. Move to index 2: Check 3. It's not 10. Move to index 3: Check 10. It matches the target value. The search stops, and the index 3 is returned.	1 mark for example (array or list) 1 mark for illustration	2
		iii	Linear search takes a number of steps that is directly proportional to the size of the list. For example, if the list has 10 elements, it might take up to 10 steps to find the target value (in the worst case). If the list has 100 elements, it might take up to 100 steps. This is why linear search is less efficient for very large lists.		1
Total:					**6**
4	a	i	Plagiarism occurs when someone uses someone else’s work or ideas without giving proper credit. By copying the paragraphs directly from the article without quotation marks or a citation, Sarah is presenting someone else’s work as her own, which is unethical and against academic rules.		1
		ii	To avoid plagiarism, Marisa should have either: • Put the copied text in quotation marks and cited the source properly, or • Paraphrased the information in her own words and still credited the original author.	1 mark for each way	2
		iii	In a school or university setting, the student could face penalties ranging from a lower grade on the project to expulsion, depending on the institution’s academic integrity policy.		1
	b		Examples of Computer Crimes that Threaten Data Privacy: **Hacking:** Hackers infiltrate computer systems or networks to steal or alter confidential data. For example, a hacker breaking into a company’s database and stealing customer credit card information is a serious breach of privacy.	1 mark for each computer crime	2

			Phishing: Cybercriminals send fake emails or messages pretending to be legitimate organizations to trick people into providing personal information like passwords or bank details. For instance, a phishing email may look like it's from a bank asking the recipient to confirm their account details, which the attacker then uses fraudulently.		
Total:					**6**
5	a			½ mark for each correct symbol	3
	b		**Top-Down Approach**: The system design begins with the high-level requirements, which are then broken down into smaller components. **Bottom-Up Approach**: Design starts with building and testing small, reusable components, which are later combined to form a complete system.		1
	c		i. Top-Down ii. Bottom-Up iii. Top-Dow iv. Bottom-Up	½ ½ ½ ½	2
Total:					**6**

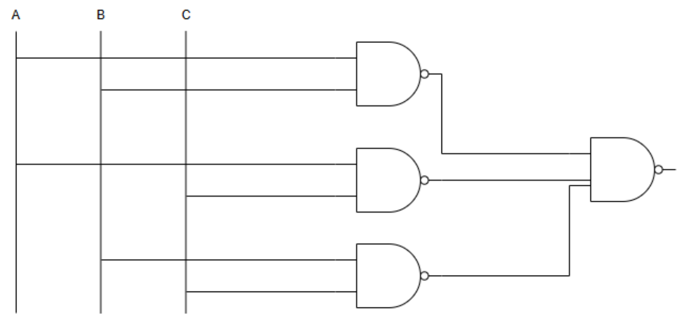
6	a		Accept replies related to: simplicity and ease of setup, centralised management, fault isolation, scalability, Reduced signal interface	1 mark for each	2
	b		In a bus topology, all devices share a single communication line. This can cause congestion if too many devices try to communicate at once. The star topology is more reliable because it isolates devices and reduces the chance of network-wide failures.		1
	c		i		1
	d	i	Packet Switching breaks data into small packets and sends them individually across the network. Need not mention that: Each packet may take different routes to reach the destination, where they are reassembled.		1
		ii	**Efficiency**: PANs often involve devices like smartphones, tablets, or wearables that communicate intermittently. Packet switching allows data to be transmitted as needed without dedicating a full communication path. **Flexibility**: Devices in a PAN can send small packets of data to each other, with minimal overhead and more efficient use of bandwidth.		1
Total:					**6**
7	a	i	Holds the address of the next instruction to be fetched from memory. or It is updated after each fetch to point to the next instruction		1
		ii	The MAR holds the address of the memory location from which data (in this case, the instruction) is to be fetched. or The IP/PC value is copied into the MAR, indicating which memory address the CPU should access to fetch the instruction.		1
		iii	The MDR holds the data read from memory or the data to be written to memory. or Once the instruction is fetched from the address in memory, it is placed into the MDR.		1
		iv	The instruction fetched from memory is moved from the MDR into the CIR. This is the register that holds the current instruction that is to be decoded and executed.		1

	b	i	Uni-directional		1
		ii	2^{32} memory locations		1
Total:					**6**
8	a		Multi-Tasking Operating System		1
	b		CPU scheduling is the process of determining which process or thread the CPU should execute next from the ready queue.		1
	c		A process holding one resource may be requesting a resource currently being held by another process, which in turn requires a resource currently held by the first process. Hence both processes are caught in a deadlock.		1
	d		In **polling** CPU keeps on checking I/O devices at regular interval whether it needs CPU service whereas, in **interrupt handling**, the I/O device interrupts the CPU and tell CPU that it needs CPU service	1 1	2
	e		It prevents processes from accessing space allocated to other jobs.		1
Total:					**6**
9	a		Helps ensure that the system is built to address the correct issues and meet stakeholder needs		1
	b		technical, economic, operational, and/or legal feasibility	1/2 mark for check	1
	c		Determine whether the proposed system can be successfully developed and implemented within constraints like budget, resources, and time.	1 for each feasibility	2
	d		Phased		1
	e		May mention: Minimize Disruption, Testing and Adaptation, Risk Management, Scale		1
Total:					**6**
10	a	i	They uniquely identify records, this is key to data integrity and efficient data access.		1
		ii	TaxiID in Rides Table		1
		iii	Drivers Table. The DriverID in the TaxisTable would be the Foreign Key linking to the Primary key in Drivers table.		2
	b		Data Security and Management Data Quality Management		2
Total:					**6**

SECTION 'B'

<table>
<tr><th colspan="3">Question No.</th><th>Solution</th><th>Mark Distribution</th><th>Marks</th></tr>
<tr><td>1</td><td>a</td><td></td><td>
<table>
<tr><th>A</th><th>B</th><th>C</th><th>F</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>1</td></tr>
</table>
</td><td>1 mark for correct inputs

2 marks for correct output</td><td>3</td></tr>
<tr><td></td><td>b</td><td></td><td>
<table>
<tr><th>C\AB</th><th>00</th><th>01</th><th>11</th><th>10</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>1</td><td>1</td></tr>
</table>
AB + AC + BC
</td><td>1 mark for correct KM

1 mark for correct grouping
1 mark for correct expression</td><td>3</td></tr>
<tr><td></td><td>c</td><td></td><td>


</td><td>NB: award 1 mark if the student got an incorrect boolean statement but implemented the correct circuit</td><td>2</td></tr>
<tr><td></td><td>d</td><td></td><td>
Data Types:

i. Unique post ID: String. A String is suitable if you want to manage and display a simple, unique identifier.
ii. Timestamp: Accept any reasonable representation such as String, long, or LocalDateTime from java.time package.
</td><td>½ mark for each datatype

½ mark for each explanation</td><td>4</td></tr>
</table>

			iii. **Likes count:** int. Since likes are always positive whole numbers, int is memory-efficient for counting likes. iv. **Text content:** String. Java Strings are used for text data and are suitable for post content.		
	e		i. Since only limited to arrays, an **array of objects** can be created where each element in the array is a Post object. Each Post object would contain fields for the unique ID, timestamp, likes count, and content. ii. **Example:** Post [] userPosts = new Post [100]; This array would hold up to 100 posts for the user. Each Post in userPosts could store all necessary information for a single post, allowing easy access to each post's details by accessing individual elements in the array, e.g., userPosts[0], userPosts[1], etc. NB: accept also ArrayList if candidtates opt for it.		1 1
	f		Any TWO from: • Stack • Queues Linear Lists	1 mark for each type	2
	g	i	public double calculateAverageLikes(Post[] posts); NB: accept also ArrayList if students opt for it.		1

		ii	Example code:
			public double calculateAverageLikes(Post[] posts) {
			  int totalLikes = 0;
			  for (Post post : posts) {
			    totalLikes += post.getLikes(); // Assuming getLikes() returns the number of likes
			  }
			  return (double) totalLikes / posts.length;
			}
				1 for totalLikes init
				
				1 for loop and increment of totalLikes
				
				1 for calculation and return
					3

Total:					**20**
2	a		- LLL are more difficult to code in and debug - HLLs are more portable - HLLs have more statement power	1 each	2

	b		This is because HLLs are not native to the CPU, which is binary in nature		1
	c		Compilers are platform-specific and so the object code produced would run on a given platform and refer to a given instruction set.		1
	d		Interpreter. An interpreter is used during runtime to execute code line by line, making it easier to identify and debug errors as they occur.	1 1	2
	e		Interpreter		1
	f	i	Combining the object code into a single executable program.		1
		ii	Loading programs into memory and preparing them for execution.		1
	g	i	A two-dimensional array called salesData. data of type integer		1
		ii	Two more rows added to the array In line 10, 3 should be changed to 5	1 1	2
		iii	totalSales[i] += salesData[i][j];		1
		iv	mostProfitable = i;		1
		v	System.out.println("Most Profitable Branch: Branch " + (mostProfitable + 1)); System.out.println("Total Sales: €" + maxSales);	1 1	2
		vi	In its current form, the code is more procedural/imperative in style. This is due to the logic residing in a single method (main). Accept any reasonable improvements proposed by the student that would make the program more object-oriented. For example, encapsulating sales data and related operations into a Branch class; separating responsibility for the program into multiple classes, etc.	1 1	2
	h		May discuss points related to: - **Data Protection**: Prevents unauthorized or unintended access and modification of sensitive data. (Exemplification: access modifiers such as private or protected can be used to protect data). - **Integrity**: Enforces business rules, validation, and audits through controlled access. (Exemplification:	½ for each explanation ½ for each reasonable exemplification of the discussion	2

			A Branch can enforce validation for sales data updates.) - **Separation of Concerns**: Reduces global dependencies and limits potential attack vectors. (Exemplification: Separating the program into multiple classes, each with their own concern such as Branch, BranchManager, Employee, etc). - **Access Control**: Allows fine-grained control over who can access or modify the data. (Exemplification: We may not want all employees to gain access to sales data, so only managers would be allowed to.)	related to the source code	
Total:					**20**