

2023 JC2 Prelim Paper 1 - Suggested Marking Scheme

Qn	Task
1a	<pre> classDiagram class Equipment { -model_no -quantity -rental_cost -num_days_rented +get_model_no() +set_model_no() +get_quantity() +set_quantity() +get_rental_cost() +set_rental_cost() +get_num_days_rented() +set_num_days_rented() +calculate_equipment_rental() } class Photography_Equipment { -storage_device_included -tripod_included +get_storage_device_included() +set_storage_device_included() +get_tripod_included() +set_tripod_included() } class Audio_Equipment { -av_technician_name -setup_venue +get_av_technician_name() +set_av_technician_name() +get_setup_venue() +set_setup_venue() } Equipment < -- Photography_Equipment Equipment < -- Audio_Equipment </pre>
	Appropriate class names (singular form)
	Equipment private attributes - model_no - quantity - rental_cost - num_days_rented
	Equipment getters and setters for attributes (public)
	method to calculate cost of equipment rental (public)
	PhotographyEquipment and AudioEquipment inherit from Equipment
	AudioEquipment private attributes - storage_device_included - tripod_included
	PhotographyEquipment private attributes - av_technician_name - setup_venue

	AudioEquipment & PhotographyEquipment getters and setters for attributes (public)
	Penalties
1b	
	<u>The purpose of a superclass is to promote code reuse by allowing subclasses to access public methods of a parent class.</u>
	PhotographyEquipment and AudioEquipment are subclasses of the superclass Equipment . They both inherit properties from Equipment , such as <code>get_model_no()</code> , <code>set_model_no()</code> etc. without the need to re-implement/duplicate those methods in their own classes.
1c	
	<u>Encapsulation protects the data against inadvertent modification.</u>
	It also separates the interface of the object or class from its implementation .
	Penalties
1d	
	In the PhotographyEquipment class, a private attribute isPhotographerNeeded (True/False) and its public getter, setter methods
	as well as a public method calculate_equipment_rental() that overrides the method of the same name in Equipment , as it uses a different formula to compute the rental cost.
1e	
	<u>Polymorphism promotes code generalisation.</u>
	as polymorphic classes can be used interchangeably without requiring conditional code ; eg. <code>calculate_euipment_rental</code> method can be added to the subclasses, without modifying the superclass.
2a	
	Consistency and Standards <i>(or another appropriate usability principle)</i>
	The links are located in a particular section of the page and it remains the same across the pages. It enhances the user experience by promoting access, in such a way that they would not have to frustratingly locate the links at different parts on each page. <i>(or another appropriate description)</i>
2b	<i>Other suitable answers accepted</i>
(i)	At point 2, the user (presumably the inventory manager) is asked to enter the quantity of products sold. It may not be a real-time update.
	During this time, additional products could have been sold,
	meaning an outdated value may be used for subtracting products from the current inventory level in point 3, giving rise to inaccurate inventory levels.
(ii)	Replace point 2 with, "Upon completion of the sale,
	automatically register the quantity of products sold"

3a	<pre> graph TD Melon((Melon)) --- Guava((Guava)) Melon --- Olive((Olive)) Guava --- Apple((Apple)) Guava --- Lemon((Lemon)) Lemon --- Mango((Mango)) Olive --- Peach((Peach)) Peach --- Papaw((Papaw)) Peach --- Prune((Prune)) </pre>																																		
(i)	up to 3 marks for correct BST (left side < node, right side > node)																																		
	1 additional mark for correct preorder (exactly as shown)																																		
(ii)	Apple, Guava, Lemon, Mango, Melon, Olive, Papaw, Peach, Prune																																		
3b	<table border="1"> <thead> <tr> <th>Index</th><th>Key</th><th>Value</th></tr> </thead> <tbody> <tr><td>0</td><td>Peach</td><td>1</td></tr> <tr><td>1</td><td>Prune</td><td>1</td></tr> <tr><td>2</td><td>Papaw</td><td>1</td></tr> <tr><td>3</td><td>Mango</td><td>1</td></tr> <tr><td>4</td><td></td><td></td></tr> <tr><td>5</td><td>Melon</td><td>1</td></tr> <tr><td>6</td><td>Guava</td><td>1</td></tr> <tr><td>7</td><td>Apple</td><td>1</td></tr> <tr><td>8</td><td>Lemon</td><td>1</td></tr> <tr><td>9</td><td>Olive</td><td>1</td></tr> </tbody> </table>	Index	Key	Value	0	Peach	1	1	Prune	1	2	Papaw	1	3	Mango	1	4			5	Melon	1	6	Guava	1	7	Apple	1	8	Lemon	1	9	Olive	1	
Index	Key	Value																																	
0	Peach	1																																	
1	Prune	1																																	
2	Papaw	1																																	
3	Mango	1																																	
4																																			
5	Melon	1																																	
6	Guava	1																																	
7	Apple	1																																	
8	Lemon	1																																	
9	Olive	1																																	
(i)	<i>Keys inserted in the correct order</i>																																		
	<i>Values assigned correctly</i>																																		
(ii)	In order to store more than ten entries, the current hash table is not able to do so. A new hashtable with a larger capacity has to be created, and																																		
	the current entries have to be extracted, rehashed (using the current or a new hash function) with the new size and inserted into the new hash table <i>Other suitable answers accepted.</i>																																		

3c	
(i)	1. $O(\log n)$ if the binary search tree is balanced and degrades to $O(n)$ if the tree is maximally unbalanced 2. $O(1)$, for a hash table with no prior collisions
(ii)	Since there are a large variety of fruits, the chances of hash collisions could increase for the hash table, meaning that the collision resolution (open or closed addressing), or the need for hash table resizing could increase the time complexity for update(of count) from $O(1)$ to $O(n)$.
	However, for the BST, the search and update(of count) would be $O(\log n)$ for a balanced tree, degrading to $O(n)$ only for grossly unbalanced trees, in which case it could probably undergo rebalancing. Given the time complexity comparison, the BST would be more appropriate for this scenario. <i>Alt answer in favour of the hash table:</i> if candidate suggests a way to have the data structures stay close to optimal (with relevant discussion of time complexity), hash table is a reasonable choice as well
4a	
	For a recursive function to be successful, it should handle a base case that does not require further processing ,
	It should call itself within its instructions ,
	and each recursive function call should bring the problem closer to the base case .
4b	
(i)	The program will face a runtime error, as it goes into an infinite loop.
(ii)	The error happens as a) the program does not reach the terminating condition where the denary_value ≤ 0 , b) because the variable denary_value is not decremented between lines 10 and 11.
(iii)	Between lines 10 and 11: denary_value \leftarrow denary_value DIV 2 <i>Note: Line 10 should be indented (no marks deducted, as it is an errata in the paper)</i>
(iv)	denary_value binary_string remainder
	9 ' ' '1'
	4 '1' '0'
	2 '01' '0'
	1 '001' '1'
	0 '1001'

(v)	<p><i>Suggested answer:</i></p> <pre> FUNCTION denary_to_binary_recursive(denary_value: INTEGER) RETURNS STRING IF denary_value <= 1 THEN RETURN 'STRING(denary_value)' ENDIF remainder ← STRING(denary_value MOD 2) denary_value ← denary_value DIV 2 RETURN CONCAT(denary_to_binary_recursive(denary_value), remainder) </pre> <p><i>OR</i></p> <pre> FUNCTION denary_to_binary_recursive(denary_value: INTEGER) RETURNS STRING remainder ← STRING(denary_value MOD 2) denary_value ← denary_value DIV 2 IF denary_value = 0 THEN RETURN remainder ENDIF RETURN CONCAT(denary_to_binary_recursive(denary_value), remainder) </pre>
	appropriate base case present
	remainder and denary_value correctly calculated
	recursive function call properly placed within the concatenation function and returned
(vi)	<p>↑ return '1001'</p> <p>denary_to_binary(9)</p>
	<p>↓ ↑ return '100'</p> <p>denary_to_binary(4)</p>
	<p>↓ ↑ return '10'</p> <p>denary_to_binary(2)</p>
	<p>↓ ↑ return '1'</p> <p>denary_to_binary(1)</p>
5a	
	Purpose Limitation Obligation requires MyJobPortal to collect, use or disclose personal data for purposes that a reasonable person would consider appropriate under given circumstances,
	and for which the individual has given consent to.
	MyJobPortal should actively track the uses of the data,
	by implementing logging system (Accounting under the AAA framework), or authentication system to identify users who use the data
5b	
	Protection obligation requires MyJobPortal to make reasonable security arrangements to protect the personal data in their possession to
	prevent unauthorised access, collection, use, disclosure or similar risks.
	MYJobPortal should implement an intrusion prevention system (IPS) that

	<p>actively takes steps to prevent intrusion or an attack when it identifies one, reducing the risk of the personal data being exposed to malicious users.</p> <p><i>Other possible answers from syllabus: Authorisation (e.g. access control system), with elaboration...</i></p>
5c	
	Data Breach Notification Obligation requires MyJobPortal to notify PDPC and affected individuals as soon as practicable,
	in the event of a data breach that likely results in significant harm to individuals, and/or are of significant scale.
	<p>MyJobPortal should keep detailed records of system and network logs to</p> <ul style="list-style-type: none"> - identify the scale of the damage (locations and extent of breach) - make restitution to stop/prevent the breach - share the information with the authorities <p><i>This is also a place where intrusion detection system would be an obvious measure to implement</i></p>
6a	
(i)	A local area network comprises devices connected within a small geographical area, typically within a room or building,
	whereas a wide area network is a network of computing devices covering a large-scale geographical area, typically across multiple geographical locations.
(ii)	There is no necessity for a dedicated transmission path here as data is not time-sensitive (unlike video/audio streaming), or there might not be sufficient network resources to dedicate a transmission path.
	Therefore, the computers and printers use a packet switched network, where the data is broken up into packets and each packet may go by a different (optimal) route to reach the printer.
6b	
(i)	Internet Layer
(ii)	Application Layer
6c	
(i)	1. Range Check (most appropriate answer)
	2. Format Check (most appropriate answer)
(ii)	<p>TCP uses the Checksum data verification technique.</p> <p>For general info: UDP supports Checksum as well.</p>
	<p>TCP helps in ensuring the data is successfully received by the recipient (by requiring acknowledgement). In this context, the emphasis is on the confirmation that the print job was received by printer (e.g. so that user does not send print job twice)</p> <p><i>contextualise your answer</i></p>
7a	

	1NF requires that there are no repeated columns and all columns contain atomic data. The above table has no repeated columns: there are three different ID columns, and columns for Patient Name, Patient Contact, Appt Date, Appt Time, Doctor Name, Qual Type and Consult Fee, with each representing a different kind of data.
	Each column also contains only a single value : either a single integer or text. Therefore the table is in 1NF.
7b	
	Reducing data redundancy reduces insert, update, and delete anomalies which would lead to data inconsistencies, compromising data integrity.
	It also saves memory space of the database since it is smaller in size.
7c	
(i)	3NF requires that the table fulfils the requirements for 2NF , and each non-key column should be non-transitively dependent on the primary key .
	Here, the Qual Type and Consult Fee are transitively dependent on the primary key Doctor ID via Qual ID . Therefore, the table Doctor is not in third normal form.
(ii)	Referential integrity could be violated as the Appointment table has a foreign key reference to the Doctor table
	Data could become inconsistent , as records in the table would still have reference to the doctor's previous id // Data in the Appointment table could become orphaned <i>or any other suitable answer</i>
7d	Doctor (Doctor ID , Doctor Name, Qual ID*) <i>use a dotted line for the foreign key</i> Qualification (Qual ID , Qual Type, Consult Fee)
	Doctor table has Doctor ID [PK]
	Doctor table has Qual ID [FK]
	Qualification table has Qual ID [PK]
	Columns appropriately allocated
7e	
	Doctor ID, Patient ID, Appt Date [Composite PK]
7f	<pre> erDiagram Patient --} Appointment : "" Doctor --} Appointment : "" Doctor } --} Qualification : "" </pre>
	Patient <-- 1 -- n --> Appointment (1 to many relationship)
	Doctor <-- 1 -- n --> Appointment (1 to many relationship)
	Doctor <-- n -- 1 --> Qualification (many to 1 relationship)
7g	
	Add an additional column " Attended "
	to the table " Appointment ", to keep track of patient attendance
7h	<i>There is more than one possible answer for this question</i>

	SELECT "Doctor"."Doctor Name", "Qualification"."Qual Type", "Qualification"."Consult Fee"
	FROM "Patient"
	INNER JOIN "Appointment"
	ON "Patient"."Patient ID" = "Appointment"."Patient ID"
	INNER JOIN "Doctor"
	ON "Doctor"."Doctor ID" = "Appointment"."Doctor ID"
	INNER JOIN "Qualification"
	ON "Qualification"."Qual ID" = "Doctor"."Qual ID"
	WHERE "Patient"."Patient Name" = 'Robert'
	GROUP BY "Doctor"."Doctor Name"
	ORDER BY "Qualification"."Consult Fee" DESC
	Penalties