Higher
Coursework
Assessment Task

Higher Computing Science
Assignment
Finalised Marking Instructions

Marking instructions

General marking principles

Always apply these general principles. Use them in conjunction with the specific marking instructions, which identify the key features required in candidates' responses.

(a) Always use positive marking. This means candidates accumulate marks for the demonstration of relevant skills, knowledge and understanding; marks are not deducted for errors or omissions.

(b) If a candidate response is not covered by either the principles or specific marking instructions, and you are uncertain how to assess it, you must seek guidance from your team leader.

(c) Award marks regardless of spelling, as long as the meaning is unambiguous and does not result in a syntax error in implemented code.

(d) For design and implementation tasks, a sample response may be shown in the detailed marking instructions. This will not be the only valid response. You must use the detailed marking instructions and additional guidance to ensure that you consider alternative approaches and nuances of different programming languages. If in doubt you should refer to your team leader.

(e) If a candidate puts a score through their entire response to a question and makes a further attempt, you should only mark the further attempt. If no further attempt is made and the original is legible, you should mark the original response.

(f) In the detailed marking instructions, if a word is underlined then it is essential; if a word is in brackets() then it is not essential. Words separated by / are alternatives.

Specific marking instructions

Task	Expected response	Additional guidance	Marks available	
	Software design and development			
1a	Award 1 mark for each bullet: ♦ find the name of the attraction(s) with the highest number of visitors ♦ calculate the number of days until next roller coaster service ♦ find roller coasters which are due to be serviced within 7 days	Read from file is an input. Write to file is an output. Ignore if included in addition to the three bullet points. Must mention roller coaster in bullet points 2 and 3.	3	Analysis
1b	Award 1 mark for each bullet: ♦ attraction(),visitors() passed IN to step two with nothing passed out ♦ attraction(), category(), daysOpen() passed IN to step three with nothing passed out	Must indicate an array (). Array names and variable names must match the names in the other modules of the design. Award 0 for a bullet point if there is any additional data flow. Maximum 1 mark if bullet points 1 and 2 correct but additional data flow given in step 1.	2	Design

Task	Expected response	Additional guidance	Marks available	
	Software design and development			
1c(i)	Read in attraction data (2) ♦ module with correct parameters passed or returned ♦ assigned to five parallel arrays Find least and most visited (4) ♦ module with attraction() and visitors() passed in ♦ initialise minimum and maximum value or position ♦ find minimum and maximum value or position ♦ traverse array to display the names of the least visited (Beaver Falls, CandyFloss Carousel) and most visited (Thundergun Express) attractions Write to file (4) ♦ module with attraction() and category() and daysOpen() passed in ♦ create/open, write to and close service.csv file ♦ use of modulus to calculate the number of days until next service ♦ two if conditions met ○ = rollercoaster ○ <=7 Implementation (2) ♦ modular program matches top-level design (three sub-programs) and refinements ♦ program is maintainable	If candidate passes in array length variable in each sub-routine award 0 marks for first instance. If candidate uses array of records award 0 marks for Read procedure then accept appropriate parameter passing for remaining procedures/ functions. Do not penalise separate find min/max subroutines with attraction() and visitors() at this point. Award bullet point 3 (loop, If, assignment correct) even if bullet point 2 is incorrect and output may be wrong. Award 0 marks for bullet point 3 if pre-defined function is used. (Asteroid Belt, G-Force, Sonic Boom, Vortex) Do not penalise if additional functions are called from within step two (find min/max). Meaningful variable/sub-program names and internal commentary relevant to task.	15	Implementation

<table>
<tr><th>Task</th><th>Expected response</th><th>Additional guidance</th><th>Marks available</th><th></th></tr>
<tr><td></td><td colspan="2">Software design and development</td><td></td><td></td></tr>
<tr><td>1c(ii)</td><td>Award 1 mark for each bullet:

Count and display (3)
♦ module with height() passed in
♦ identify height of 1m or over
♦ initalise and increment count(16)</td><td></td><td>3</td><td></td></tr>
<tr><td>1d</td><td>Award 1 mark for each correct column

Sample answer<table><tr><th></th><th>If current category is ‘Roller Coaster’</th><th>days</th><th>If (90 – days) is less than or equal to 7</th></tr><tr><td>1st iteration</td><td>true</td><td>65</td><td>false</td></tr><tr><td>2nd iteration</td><td>false</td><td></td><td></td></tr><tr><td>3rd iteration</td><td>true</td><td>83</td><td>true</td></tr></table></td><td>Accept other expression of true (yes/1) and false (no/0)

Accept repetition of 65 and false for second iteration</td><td>3</td><td>Testing</td></tr>
<tr><td>1e</td><td>Award 1 mark for bullet point from data structure:
♦ array size is fixed (26 elements) and would need to be updated.
OR
♦ the array is initialised with no size and new elements are appended/arrays are redimensioned

Award 1 mark for bullet point from loops:
♦ number of iterations of loops are fixed (26)
OR
♦ the loop iterates to the length of the array/end of file</td><td>Maximum 2 marks.

Candidate response must correspond with their code.</td><td>2</td><td>Evaluation</td></tr>
</table>

Task	Expected response	Additional guidance	Marks available	
	Database design and development			
2a	Award 1 mark for each bullet. Maximum 2 marks. ♦ (A query to) sort gnomes by popularity ♦ (A query to) search for order details (including date and total cost) ♦ (A query to) search for customers who have placed orders in a given month ♦ (A query to) search for customers to contact regarding special offers ♦ (A query to) update, insert or delete data ♦ (A query to) calculate the total cost of an order	Functional requirements should be extracted from end-user information. Award a maximum of 1 mark for update, insert or delete example.	2	Analysis

2b	Award 1 mark for each bullet.
	
	♦ sum on quantity
	♦ two tables and equijoin and fields
	♦ wildcard before and after 'solar'
	♦ alias & GROUP BY
	♦ ORDER BY
	
	`SELECT gnomeName, SUM(quantity) AS [Total Gnomes Sold]`
	`FROM Gnome, GnomePurchase`
	`WHERE Gnome.gnomeID=GnomePurchase.gnomeID AND Description Like "*solar*"`
	`GROUP BY gnomeName`
	`ORDER BY SUM(quantity) DESC;`
		Accept `GROUP BY gnomeID`	5	Implementation

Task	Expected response	Additional guidance	Marks available	
	Database design and development			
2c	Award 1 mark for each bullet. ♦ MAX to find most expensive gnome ♦ use of first Query ♦ use of 2 conditions o unitPrice = [Expensive] o quantity >= 3 (>2) ♦ four tables, equijoins and fields	Query using MAX could be a sub-query. Award 0 marks for bullet 3 if unitPrice value is used instead of field.	4	Implementation

	`SELECT max(unitPrice) AS [Expensive]`
	`FROM Gnome;`
	
	`SELECT emailaddress, Orders.orderID, quantity`
	`FROM Customer, GnomePurchase, Gnome, CustOrder, MaxGnomePrice`
	`WHERE Customer.customerId=Orders.customerID AND Orders.orderID=GnomePurchase.orderID AND Gnome.gnomeID=GnomePurchase.gnomeID And unitPrice=[Expensive] And quantity >=3;`			

2d	Award 1 mark for each bullet. ♦ include SUM() aggregate function on calculation ♦ GROUP BY	Accept `GROUP BY customerID` Accept no `GROUP BY` in SQLite	2	Testing

	`SELECT forename, surname, SUM(unitPrice*1.2*quantity) AS [Total to Pay £]`
	`FROM Customer, Gnome, GnomePurchase, CustOrder`
	`WHERE CustOrder.orderID="ord0024" AND Customer.customerID = CustOrder.customerID AND CustOrder.orderID=GnomePurchase.orderID AND gnome.gnomeID=GnomePurchase.gnomeID`
	`GROUP BY forename, surname;`			

Task	Expected response	Additional guidance	Marks available	
	Database design and development			
2e(i)	Award 1 mark for each bullet. Maximum 1 mark. ♦ unit price has changed ♦ VAT has been included/changed ♦ a discount voucher was applied		1	Evaluation
2e(ii)	A field could be added that stores order total, or price when ordered.	Answers should relate to the structure of the database. Not accepting keep track of, print out.	1	

Task	Expected response	Additional guidance	Marks available	
	Web design and development			
3a	Award 1 mark for each bullet: ♦ a form allowing users to submit their own review ♦ an element that is clicked to show/hide reviews or JavaScript to show/hide reviews	Must indicate a feature of the software. Accept button, link, image.	2	Analysis
3b	Award 1 mark for each bullet: ♦ all six elements present with submit button ♦ validation of age and rating	Validation of age and rating could be radio buttons, numeric inputs or drop down list.	2	Design
3c	Award 1 mark for each bullet: ♦ ids assigned to sections and all four sections set to display:none ♦ sections revealed using onclick event on each image element ♦ function(s) to display one element while hiding others	Candidate may use their own ID and function names. Code can be HTML or CSS.	3	Implementation
3d	Award 1 mark for each bullet: ♦ radio buttons for age ♦ numeric input for rating field with correct min of 1 and max of 10 ♦ text input and textarea with correct maxlength ♦ all fields required and matching side by side design	15,15,20,4x50 Accept various implementation for side by side layout.	4	implementation

Task	Expected response	Additional guidance	Marks available	
	Web design and development			
3e	Award 1 mark for each bullet. Maximum 2 marks. ♦ Text input boxes tested with o valid number of characters o exceeding the number of characters o leaving blank ♦ Numeric field values o in range 1-10 o outwith range 1-10 o leaving blank o text ♦ Radio buttons checked for o single selection o multiple selection o left blank ♦ Text area tested with o 1 to 200 number of characters o leaving blank	Need at least two test criteria for each element for mark.	2	Testing
3f	Award 1 mark for each bullet. Maximum 2 marks. Not fit for purpose because ♦ unable to leave comments about existing reviews ♦ not all links in Navigation bar work Fit for purpose because: ♦ home page contains relevant information ♦ form included on review page ♦ JavaScript for show/hide	Accept answers referring to candidate's own implementation e.g. JavaScript function not working.	2	Evaluation

	Marks Available	Marks Awarded
Assignment total	**40**	

Task 1 – Software Design and Development		Marks Available	Marks Awarded
1a – Analysis	Attraction(s) with the highest number of visitors	1	
	Number of days until roller coaster service	1	
	Roller coasters due to be serviced within 7 days	1	
			/3

1b – Design	IN: attraction(), visitors()	1	
	IN: attraction(), category(), daysOpen()	1	
			/2

1c(i) – Implementation	Read in attraction data	Module with parameters	1	
		Assigned to five parallel arrays	1	
	Find least and most visited	Module with parameters	1	
		Initialise minimum and maximum	1	
		Find minimum and maximum	1	
		Traverse array to display	1	
	Write to file	Module with parameters	1	
		Create/open, write to and close service.csv file	1	
		Modulus	1	
		Two conditions = roller coaster, <=7	1	
	Implementation	Modular and matches top-level design	1	
		Maintainable	1	
				/12

1c(ii) – Count height	Module with parameter	**1**	
	Identify height >= 1	**1**	
	Increment count	**1**	
			/3

1d – Testing	Column 1	1	
	Column 2	1	
	Column 3	1	
			/3

1e – Evaluation	Data structure	1	
	Loops	1	
			/2

Task 2 – Database Design and Development		Marks Available	Marks Awarded	
2a – Analysis	Functional requirements	2		/2
2b – Implementation	Sum on quantity	1		
	Tables, equijoin, fields	1		
	Wildcard	1		
	Alias & GROUP BY	1		
	ORDER BY	1		/5
2c – Implementation	MAX to find most expensive gnome	1		
	Use of Query 1	1		
	Use of two conditions	1		
	Tables, equijoins and fields	1		/4
2d – Testing	SUM aggregate function	1		
	GROUP BY	1		/2
2e(i)- Evaluation	Change in price	1		/1
2e(ii)- Evaluation	Store total/original price	1		/1

Task 3 – Web Design and Development		Marks Available	Marks Awarded	
3a – Analysis	Form to submit review	1		
	Element clicked/JavaScript to show/hide	1		/2
3a – Design	Six elements and submit button	1		
	Validation of age and rating	1		/2
3c – Implementation (Review page)	IDs assigned and set to display:none	1		
	Onclick in image element to reveal	1		
	Function(s) to display	1		/3
3d – Implementation (Form)	Radio buttons for age	1		
	Numeric rating with min 1 and max 10	1		
	Text input and textarea with correct maxlength	1		
	All fields required and matching side by side design	1		/4
3e – Testing	Validation 1 with two criteria	1		
	Validation 2 with two criteria	1		/2
3f – Evaluation	Fit/Not fit for purpose reasons	1		/2