Question	Answer	Marks
1	• AO2 (maximum 9 marks) • AO3 (maximum 6 marks) **Data Structures required** names shown underlined must be used as given in the scenario 2D Array or list `Evening[1:10, 1:20] / Evening[0:9, 0:19]` Variables `Counter, SeatCounter, NumSeats, Row, Column` **Requirements (techniques)** **R1** Find number of seats available for each performance and output (searching, nested iteration, output) **R2** Inputs and validates number of seats (input, iteration, and selection) **R3** Checking if seats available (selection, assignment, output with appropriate messages) ***Example 15-mark answer in pseudocode*** `// meaningful identifier names and appropriate data structures to store the data required` `DECLARE Counter, SeatCounter, NumSeats, Row, Column : INTEGER` `CONSTANT HouseFull = 200` `CONSTANT MaxRow = 10` `CONSTANT MaxColumn = 20` `SeatCounter1 ← 0 // initialise seat counter for performance 1`	**15**

Question	Answer	Marks
1	``` FOR Row ← 1 TO 10 FOR Column ← 1 TO 20 IF Evening[Row, Column] THEN SeatCounter ← SeatCounter + 1 ENDIF NEXT Column NEXT Row // validate input OUTPUT "How many seats do you want to book? 1, 2, 3 or 4 " INPUT NumSeats WHILE 1 < NumSeats OR NumSeats > 4 OR NumSeats <> ROUND(NumSeats, 0) OUTPUT "Please enter 1, 2, 3 or 4 for the number of seats " INPUT NumSeats ENDWHILE IF SeatCounter + NumSeats > 200)// check for house full THEN OUTPUT "House full" ELSE IF SeatCounter + NumSeats > 200 // checks for not enough seats THEN OUTPUT "Only ", SeatCounter + NumSeats – 200, " seats left" ELSE FOR Counter ← 1 TO NumSeats // book required number of seats for performance Evening[MOD(SeatCounter + Counter, MaxColumn), DIV(SeatCounter + Counter), MaxColumn] ← TRUE OUTPUT "Row ", MOD(SeatCounter + Counter, MaxColumn), " seat ", DIV(SeatCounter + Counter, MaxColumn)," booked" NEXT Counter ENDIF ```	

Marking Instructions in italics			
AO2: Apply knowledge and understanding of the principles and concepts of computer science to a given context, including the analysis and design of computational or programming problems			
0	**1-3**	**4-6**	**7-9**
No creditable response.	At least one programming technique has been used. *Any use of selection, iteration, counting, totalling, input and output.*	Some programming techniques used are appropriate to the problem. *More than one technique seen applied to the scenario, check the list of techniques needed.*	The range of programming techniques used is appropriate to the problem. *All criteria stated for the scenario have been covered by the use of appropriate programming techniques, check list of techniques needed.*
	Some data has been stored but not appropriately. *Any **use** of variables or arrays or other language dependent data structures e.g. Python lists.*	Some of the data structures chosen are appropriate and store some of the data required. *More than one data structure **used** to store data required by the scenario.*	The data structures chosen are appropriate and store all the data required. *The data structures **used** store all the data required by the scenario.*

Marking Instructions in italics			
AO3: Provide solutions to problems by:			
	evaluating computer systems	**making reasoned judgements**	**presenting conclusions**
0	**1-2**	**3-4**	**5-6**
No creditable response.	Program seen without relevant comments.	Program seen with some relevant comment(s).	The program has been fully commented
	Some identifier names used are appropriate *Some of the data structures used have meaningful names.*	The majority of identifiers used are appropriately named. *Most of the data structures used have meaningful names.*	Suitable identifiers with names meaningful to their purpose have been used throughout. *All of the data structures used have meaningful names.*
	The solution is illogical.	The solution contains parts that may be illogical.	The program is in a logical order.
	The solution is inaccurate in many places. *Solution contains few lines of code with errors that attempt to perform a task given in the scenario.*	The solution contains parts that are inaccurate. *Solution contains lines of code with some errors that logically perform tasks given in the scenario. Ignore minor syntax errors.*	The solution is accurate. *Solution logically performs all the tasks given in the scenario. Ignore minor syntax errors.*
	The solution attempts at least one of the requirements. *Solution contains lines of code that attempt at least one task given in the scenario.*	The solution attempts to meet most of the requirements. *Solution contains lines of code that perform most tasks given in the scenario.*	The solution meets all the requirements given in the question. *Solution performs all the tasks given in the scenario.*