**HL Unit 5** **– Abstract Data Structures**  
Quiz 1 – 2D Arrays

|  |  |  |  |
| --- | --- | --- | --- |
| **Question 1** | | | |
| Objectives: | 5.1.4, 5.1.5 | Exam Reference: | Nov-17 15.a.b.c |

The **collection** WEATHER contains the temperatures that have been measured for one city

over the course of **one week**, starting on Monday and ending on Sunday. Each day,

24 readings were taken, one each hour, the first being at 00:00, the second at 01:00 and so

on. The data is stored in chronological order with the data for Monday stored in the collection

first, followed by Tuesday and so on.

1. State the total number of readings that were taken during this week. [1]
2. Construct the algorithm to read this data into a 2D array, A, that would allow the

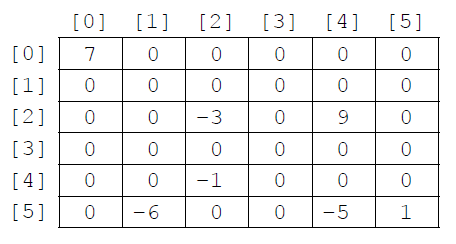
temperature on a specific day at a specific time to be accessed directly. [4]

1. Construct the algorithm that will output the day, as a word (for example Tuesday), on

which the highest temperature was recorded. [6]

|  |  |  |  |
| --- | --- | --- | --- |
| **Question 2** | | | |
| Objectives: | 5.1.4, 5.1.5 | Exam Reference: | Nov-17 14 |

Consider the following two-dimensional array, MAT, with dimensions 6 × 6.



The value −1 is stored in MAT at position [4][2]. The position [4][2] means row 4  
 and column 2.

1. State the total number of elements stored in MAT. [1]

36

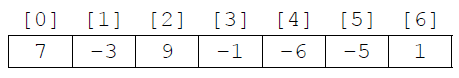
1. State the number of non-zero elements in MAT. [1]

7

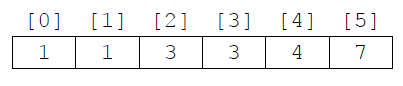
A two-dimensional array in which most of the elements are zero is called a **sparse matrix**. A sparse matrix can be compressed by storing only non-zero elements using three one‑dimensional arrays.

The **first array**, VALUES, stores all non-zero elements taken from the sparse matrix in row‑major order (left-to-right then top-to-bottom order).

The length of the array VALUES is equal to the number of non-zero elements in the sparse matrix. For the sparse matrix above, MAT, the array VALUES is:

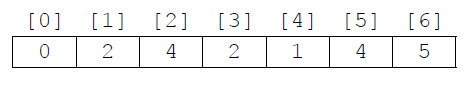


The **second array** is ROWC. ROWC[i] stores the number of non-zero elements, from row 0 to row i of the sparse matrix, **inclusive**.

The length of ROWC is equal to the number of rows in the sparse matrix. For MAT the array ROWC is

For example, ROWC[2] stores 3 because in MAT there are three non-zero elements from row 0 to row 2, inclusive.

The **third array**, COL, stores the column index for each non-zero element in the sparse matrix. COL[i] stores   
the sparse matrix column index for the non-zero element stored in VALUES[i]. For MAT the array COL is:



1. Construct an algorithm that compresses a 6 × 6 two-dimensional array, such as MAT,   
   into the three one-dimensional arrays described on page 8. You may assume that the   
   6 × 6 array is inputted and all three one-dimensional arrays are initialized. [6]

*Award marks as follows up to* ***[6 max]****. (There are 7 marking points)*

*Award* ***[1]*** *for initialization and correct changes of K (index/position in arrays*

*VALUES and COL);*

*Award* ***[1]*** *for initialization and correct changes of COUNT (counts non-zero*

*elements);*

*Award* ***[1]*** *for correct conditions in “row” loop;*

*Award* ***[1]*** *for correct conditions in “column” loop;*

*Award* ***[1]*** *for placing non-zero element at correct position in array VALUES;*

*Award* ***[1]*** *for placing the “column” index of the non-zero element at correct*

*position in array COL;*

*Award* ***[1]*** *for placing COUNT at correct position in array ROWC;*

COUNT = 0

K = 0

loop for I from 0 to 5

loop for J from 0 to 5

if MAT[I][J]! = 0 then

VALUES[K] = MAT[I][J]

COL[K] = J

K = K + 1

COUNT = COUNT + 1

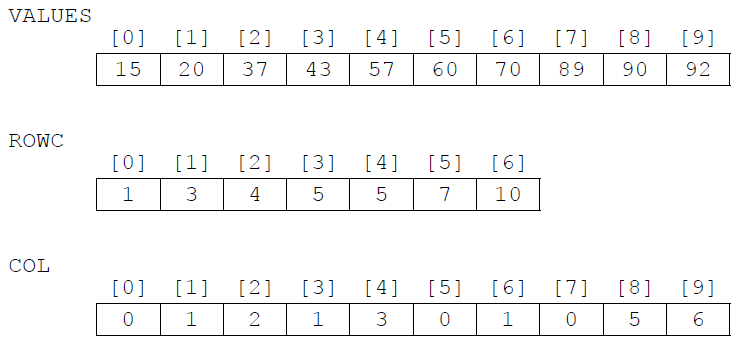
end if

end loop

ROWC[I] = COUNT

end loop

Consider the following three arrays. They hold the compressed contents of a 7 × 7 sparse   
 matrix, BIGMAT.



1. For a given column, C, in BIGMAT, outline how it could be determined that this column contains no non-zero elements. [2]

*Award up to* ***[2 max]****.*

***Example answer 1***

Array COL should be searched for (value) C;

If (value) C is not found in array COL then this column (the column whose index in

BIGMAT is C) holds only zeros;

***Example answer 2***

If the number of occurrences of (value) C in array COL;

Equals zero then this column holds only zeros;

***Example answer 3****: Award* ***[1 only]****.*

If COL[C] = COL[C–1], then the column with index C in BIGMAT contains no

non-zero-elements;

*(Accept words to that effect: “if the difference between COL[C] and COL[C-1] is*

*zero, then...”);*

1. State how many rows in BIGMAT contain only zeros. [1]

1

1. (i) State the index in VALUES of the first non-zero element in row 5 of BIGMAT. [1]

5

(ii) For a given row, R, in BIGMAT, determine the range of indexes in VALUES where  
 non-zero elements in row R of BIGMAT are placed. You may assume that there  
 is at least one non-zero element in row R. [3]

*Award marks as follows up to* ***[3 max]****.*

*Award* ***[1]*** *for realizing that the range should be determined differently for*

*the first row (when row index R is 0) OR correct range when row index is 0;*

*Award* ***[1]*** *for correct first index in range (when row index R is not 0);*

*Award* ***[1]*** *for correct last index in range;*

If row index R is equal to 0 then the range is from 0 to ROWC[0]-1 ;

If row index R is not equal to 0 then the range is from ROWC[R-1];

To ROWC[R]-1;

***Note****: Award* ***[2 max]*** *for a correct calculation of the indexes, but no*

*unifying expression showing how they have been calculated is given).*