CSE2011- Data Structures and Algorithm

Questions and Algorithms for Lab-2

1- Reverse the array of given size 'n' by using the temporary variable.

2- Write a program to print union and intersection of two array of 'n' dimension.

3- Consider two-dimensional array of MXN dimension. Insert the element in matrix and perform the following operations on array

a) Display the transpose of array

b) Calculate and display the addition of two 2D Matrix

c) Calculate and display the multiplication of 2D array

4- Consider the following scenario for a square matrix

1	2	3
4	5	6
7	8	9

Once the square matrix is rotated

i) by 90 degree in a clockwise direction then the transformed matrix will be as follows

7	4	1
8	5	2
9	6	3

ii) by 180 degree in a clockwise direction will lead to following

9	8	7
6	5	4
3	2	1

iii) by 90 degree in a anti-clockwise direction then the transformed matrix will be as follows

3	6	9
2	5	8
1	4	7

ii) Matrix rotation by 180 degree in anti-clockwise direction will lead to following

9	8	7
6	5	4
3	2	1

Write a program for the above-mentioned scenarios. (Use both brute force and in-place approach):

CSE2011- Data Structures and Algorithm

In-place algorithms: are the algorithms that does not takes extra memory in order to perform the task; except few constant additional memory requirements.

1ˢᵗ approach: using brute force approach using extra space and using the observation

Algorithm:

Rotate(Arr,dim)

Step-1 create one temporary matrix with same dimension

Step-2 for i=0 to dim

Step-3   for j=0 to dim

Step-4   ArrTemp[j][n-i-1]=Arr[i][j];

Step-5   end inner loop

Step-6 end outer loop

Step-7 Exit

2ⁿᵈ Approach: using Transpose and Rotation

Logic: Transpose ➔rotation

Algorithm:

Rotate(Arr,dim)

Step-1 for i=0 to dim

Step-2   for j=0 to dim

Step-3   Swap Arr[j][i] with Arr[i][j]//Transpose

Step-4   end inner loop

Step-5 end outer loop

Step-6 for i=0 to dim

Step-7   for j=0 to dim/2

Step-8   Swap Arr[i][j] with Arr[i][dim-j-1]

Step-9   end inner loop

Step-10 end outer loop

5- Create Sparse Matrix as Triplet representation and perform following operation

i) Print new representation of sparse matrix

ii) Addition operation of 2 sparse matrix

iii) Transpose of sparse matrix

CSE2011- Data Structures and Algorithm

Logic : count the non zero element present in the array. create one matrix of size 3X[count].

Algorithm

SparseMatrix(Arr, row,col)

Step-1 declare and initialize count

Step-2 for i=0 to row

Step-3   for j=0 to col

Step-4   if arr[i][j] is the non zero element

                              Increment count

Step-5   end inner loop

Step-6 end outer loop

Step-7 define the matrix of size 3Xsize and k=0

Step-8 for i=0 to row

Step-9   for j=0 to col

Step-10   if arr[i][j] is the non zero element

                              SparseMat[0][k]=row;

                     SparseMat[1][k]=col; SparseMat[2][k]=Arr[i][j]

                              Increment k

Step-11   end outer loop

Step-12 end inner loop

Step-13 for i=0 to 2

Step-14   for j=0 to size

                     Print SparseMat[i][j];

                 End inner loop

         End outer loop

Step-15 exit