Data Structures & Algorithms Using C

Sparse Matrix and it's Representations

Contents:

What are SPARSE MATRICES?

- A matrix is a 2-dimensional data object composed of m rows and n columns, therefore having total m x n values.
- In computer programming, a matrix can be defined with a 2-dimensional array.
  - Any array with 'm' rows and 'n' columns represent a m X n matrix.
- If most of the elements of the matrix have **0 value**, then such a matrix is termed as **"Sparse matrix"**

Consider the following as an example of a sparse matrix ***A***:

$$
\begin{bmatrix}
10 & 0 & 3 & 0 & 0 & 0 \\
21 & 2 & 0 & 4 & 0 & 0 \\
0 & 3 & 6 & 0 & 5 & 0 \\
0 & 0 & 23 & 24 & 0 & 26 \\
81 & 0 & 0 & 94 & 65 & 0 \\
71 & 92 & 0 & 0 & 55 & 36
\end{bmatrix}
$$

Why to use Sparse Matrix?

- **Storage:** There are lesser non-zero elements than zeros and thus lesser memory can be used to store only those elements.
  - space calculation: Matrix having m rows and n columns the space required to store the numbers will be m*n*s where s is the number of bytes required to store the value.

  E.g. Suppose there are 10 rows and 10 columns and we have to store the integer values then the space complexity will be bytes.

  10*10*2=200 bytes.

- **Computing time:** Computing time can be saved by logically designing a data structure traversing only non-zero elements.

Sparse Matrix Representations

- Representing a sparse matrix with a 2D array results in wastage of memory and processing time.

  E.g. consider a matrix of size 100 X 100 containing only 10 non-zero elements.
  - In this matrix, only 10 spaces are filled with non-zero values and remaining spaces of the matrix are filled with zero.
  - <u>Space allocated:</u> 100 X 100 X 2 = 20,000 bytes to store this integer matrix.
  - <u>Access time of 10 non-zero elements:</u> 10,000 scans.

- To avoid such circumstances different techniques are used such as:

  1) Triplet Representation (Array Representation)

  2) Linked list representation

Method 1: Triplet Representation (Array Representation)

- 2D array is used to represent a sparse matrix in which there are three rows named as:
- **Row:** Index of row, where non-zero element is located
- **Column:** Index of column, where non-zero element is located
- **Value:** Value of the non zero element located at index – (row , column)

Triplet as - (Row, Column, value)

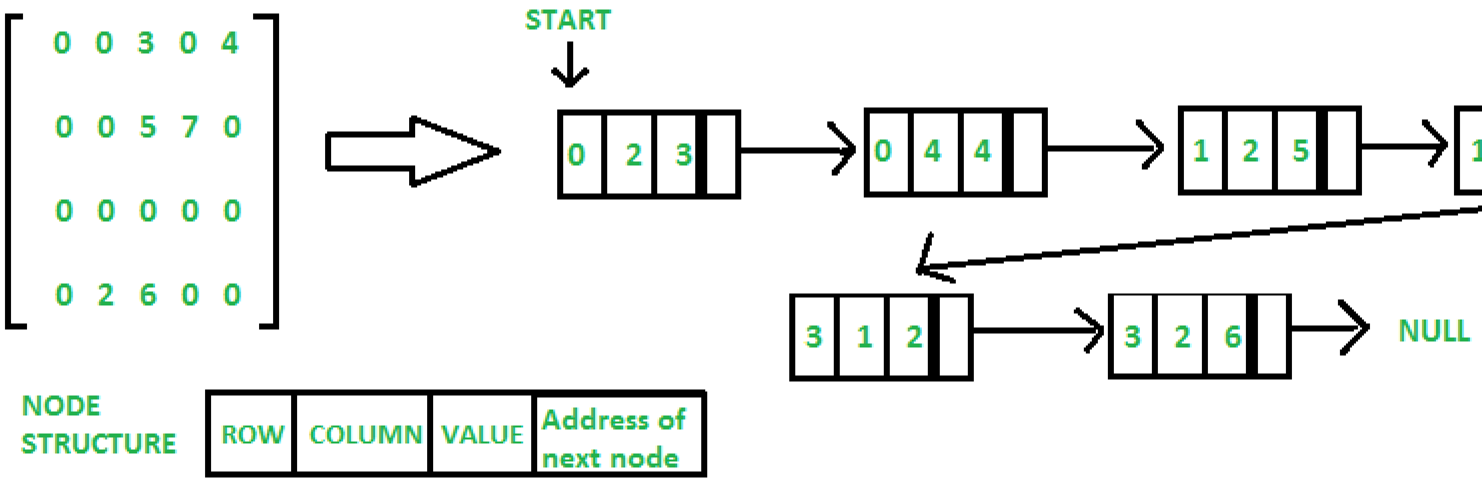
Consider the following as an example of a sparse matrix ***B***:

$$\begin{bmatrix} 0 & 0 & 3 & 0 & 4 \\ 0 & 0 & 5 & 7 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 6 & 0 & 0 \end{bmatrix} \Longrightarrow$$

Row	0	0	1	1	3	3
Column	2	4	2	3	1	2
Value	3	4	5	7	2	6

Method 2: Linked List Representation

- In linked list, each node has four fields. These four fields are defined as:
- **Row:** Index of row, where non-zero element is located
- **Column:** Index of column, where non-zero element is located
- **Value:** Value of the non zero element located at index – (row , column)
- **Next node:** Address of the next node



Classification of Sparse Matrix

Triangular Matrices

- Triangular matrices have the same number of rows as they have columns; i.e. they have n rows and n columns.
- Thus, **triangular matrix** is a special kind of square matrix.

Band Matrix

- A **band matrix** is a sparse matrix whose non-zero entries are confined to a diagonal *band*, comprising the main diagonal and zero or more diagonals on either side.

Types of Triangular Matrices

Upper Triangular Matrices

- A matrix ***A*** is an upper triangular matrix if its nonzero elements are found only in the upper triangle of the matrix, including the main diagonal;

$$U = \begin{bmatrix} u_{1,1} & u_{1,2} & u_{1,3} & \ldots & u_{1,n} \\ & u_{2,2} & u_{2,3} & \ldots & u_{2,n} \\ & & \ddots & \ddots & \vdots \\ & & & \ddots & u_{n-1,n} \\ 0 & & & & u_{n,n} \end{bmatrix}$$

Lower Triangular Matrices

- A matrix ***A*** is a lower triangular matrix if its nonzero elements are found only in the lower triangle of the matrix, including the main diagonal;

$$L = \begin{bmatrix} \ell_{1,1} & & & & 0 \\ \ell_{2,1} & \ell_{2,2} & & & \\ \ell_{3,1} & \ell_{3,2} & \ddots & & \\ \vdots & \vdots & \ddots & \ddots & \\ \ell_{n,1} & \ell_{n,2} & \ldots & \ell_{n,n-1} & \ell_{n,n} \end{bmatrix}$$

Types of Band Matrices

Diagonal Matrix

- Let A be a square matrix (with entries in any field). If all off- diagonal entries of A are zero, then A is a diagonal matrix.

Square diagonal matrix:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & -2 \end{bmatrix}$$

Rectangular diagonal matrix:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & -3 \\ 0 & 0 & 0 \end{bmatrix} \text{ or } \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 & 0 \\ 0 & 0 & -3 & 0 & 0 \end{bmatrix}$$

Tri-Diagonal Matrices

- A **tri-diagonal matrix** is a matrix that has nonzero elements only in the main diagonal, the first diagonal below this, and the first diagonal above the main diagonal.

$$\begin{pmatrix} 1 & 4 & 0 & 0 \\ 3 & 4 & 1 & 0 \\ 0 & 2 & 3 & 4 \\ 0 & 0 & 1 & 3 \end{pmatrix}.$$

References

[1] https://www.geeksforgeeks.org/sparse-matrix-representation/

[2] Lipschutz, S. (1987). *Schaum's Outline of Data Structure*. McGraw-Hill, Inc.

[3] https://en.wikipedia.org/wiki/Sparse_matrix

Thank You