**Experiment 2**

**Aim:**Implement sparse matrix using array. Description of program:

1. Read a 2D array from the user.
2. Store it in the sparse matrix form, use array of structures.
3. Print the final array.

**Algorithm:**

1. A matrix is initialized with rows and columns and a variable count is declared with count as 0.

2. List all the non-zero elements in the matrix with row and column indexing.

3. If the count is more than (row\*column)/2, the matrix is a sparse matrix.

4. Another new array with the rows equal to no. of non-zero elements in the matrix and column equal to 3 arguments is made.

5.Array is then filled with non-zero elements.

**Source Code:**

#include <stdio.h>

#include <stdlib.h>

int main() {

    int row, column, matrix[10][10], count = 0;

    printf("Number of rows: ");

    scanf("%d", &row);

    printf("Number of columns: ");

    scanf("%d", &column);

    printf("Elements of the matrix: ");

    for (int i = 0; i < row; i++)    {

        for (int j = 0; j < column; j++)        {

            scanf("%d", &matrix[i][j]);

        }

    }

    printf("Matrix is:\n");

    for (int i = 0; i < row; i++) {

        for (int j = 0; j < column; j++)        {

            printf("%d\t", matrix[i][j]);

        }

        printf("\n");

    }

    for (int i = 0; i < row; i++)    {

        for (int j = 0; j < column; j++)        {

            if (matrix[i][j] == 0)

            {

                count++;

            }

        }

    }

    if (count > ((row \* column) / 2))

        printf(" \nMatrix is a sparse matrix");

    else

        printf("\nMatrix is not a sparse matrix");

}

**Output:**

