readme

Sparse Matrix Operations

This project contains C code to perform operations on sparse matrices. A sparse matrix is a matrix in which most of the elements are zero. By contrast, if most of the elements are nonzero, then the matrix is considered dense.

Getting Started

Prerequisites

The code is written in C, so you need a C compiler to run the code. Any standard C compiler like gcc should work.

Running the Code

Clone the repository.

Open a terminal/command prompt in the directory where the code is located.

Compile the code using a C compiler. If you're using gcc, you can use the command gcc -o main main.c.

Run the executable. If you're on a Unix-like system, use ./main. If you're on Windows, use main.

Code Structure

The code uses a multi-linked circular list to represent a sparse matrix. Each node in the list can either be a head node or an entry node. A head node represents a row or a column in the matrix, and an entry node represents a non-zero element in the matrix.

The code includes the following functions:

`Read_Sparse()` : Reads a sparse matrix from the user.

`Print_Sparse(matrixPointer node)`: Prints a sparse matrix.

`Delete_Matrix(matrixPointer node)`: Deletes a sparse matrix and frees the memory.

`Transpose_Matrix(matrixPointer node)`: Returns the transpose of a sparse matrix.

`IsLowerTriangular_Matrix(matrixPointer node)`: Checks if a matrix is lower triangular.