

Assignment #2

Submission type: individual

Submission files: Zipped file (report in PDF and C codes)

Deadline: May 3, 2020 (end-of-the-day)

Programming project (Exercise 6, page 186)

Write a complete C program that performs following arithmetic operations on sparse matrices and write a report in PDF (free-form).

The report should include:

1. Development environment (e.g. OS, IDE, etc.)
2. Explanation of the algorithm and the code

mread : Read in a sparse matrix (Program 4.23, page 182-183)

mwrite : Write out a sparse matrix (Program 4.24, page 184)

merase : Erase a sparse matrix (Program 4.25, page 185)

madd : Create the sparse matrix $d = a + b$

mmult : Create the sparse matrix $d = a * b$

mtranspose : Create the sparse matrix $b = a^T$

Requirements

1. Use linked list
2. Write a menu-driven program with the same flow as the following
 - A. Start program
 - B. Print “1. Read\n2. Write\n3. Erase\n4. Add\n5. Multiply\n6. Transpose\n7. Exit\n” and wait for user input
 - C. Perform arithmetic operations
 - Menu 1: Print “Save index: ” and wait for user input for a number. Execute *mread* operation (follow the textbook implementation for printing the dialog, output, and user input). Go back to menu.
 - Menu 2: Print “Print index: ” and wait for user input for a number. Execute *mwrite* operation (follow the textbook implementation for printing the dialog and output). Go back to menu.

Menu 3: Print “Erase index: ” and wait for user input for a number. Execute *merase* operation. Go back to menu.

Menu 4: Print “First, second, save index: ” and wait for user input for three numbers separated by a space. Execute *madd* operation. Go back to menu.

Menu 5: Print “First, second, save index: ” and wait for user input for three numbers separated by a space. Execute *mmult* operation. Go back to menu.

Menu 6: Print “Transpose index, save index: ” and wait for user input for two numbers separated by a space. Execute *mtranspose* operation. Go back to menu.

Menu 7: Exit the program

3. You are encouraged to use the *mread*, *mwrite*, *merase* implementation from the textbook (uploaded in the Blackboard Course Materials). For these operations, program must print the same dialog and output, i.e. *printf*, and accept the same user input, i.e. *scanf*, as the textbook’s implementation. If you modify or write a completely different code, you must explain it in the report extensively (e.g. how it differs from the textbook implementation).
4. For *mread*, *madd*, *mmult*, *mtranspose* operations, you will save the result of the operation to a specified index. If the specified index is already occupied, you will overwrite it.
5. A matrix can be saved in any index between 1 and 64.
6. You must catch and handle most errors.

TA’s note: Beware of plagiarism (we use proprietary plagiarism checkers in addition to BlackBoard plagiarism checker). Ask the TAs if you have any questions.

2020/04/22 Edit: Added details on “saving” a matrix.