Assignment

The typical linear equation is known as $Ax = b$ form and its solution can be found with the $S = A^{-1}b$ method.

A will be assumed diagonal for the simplification matrix, which means only the diagonal elements are different from zero.

So, with that assignment, you will try to solve a linear equation by writing a C code.

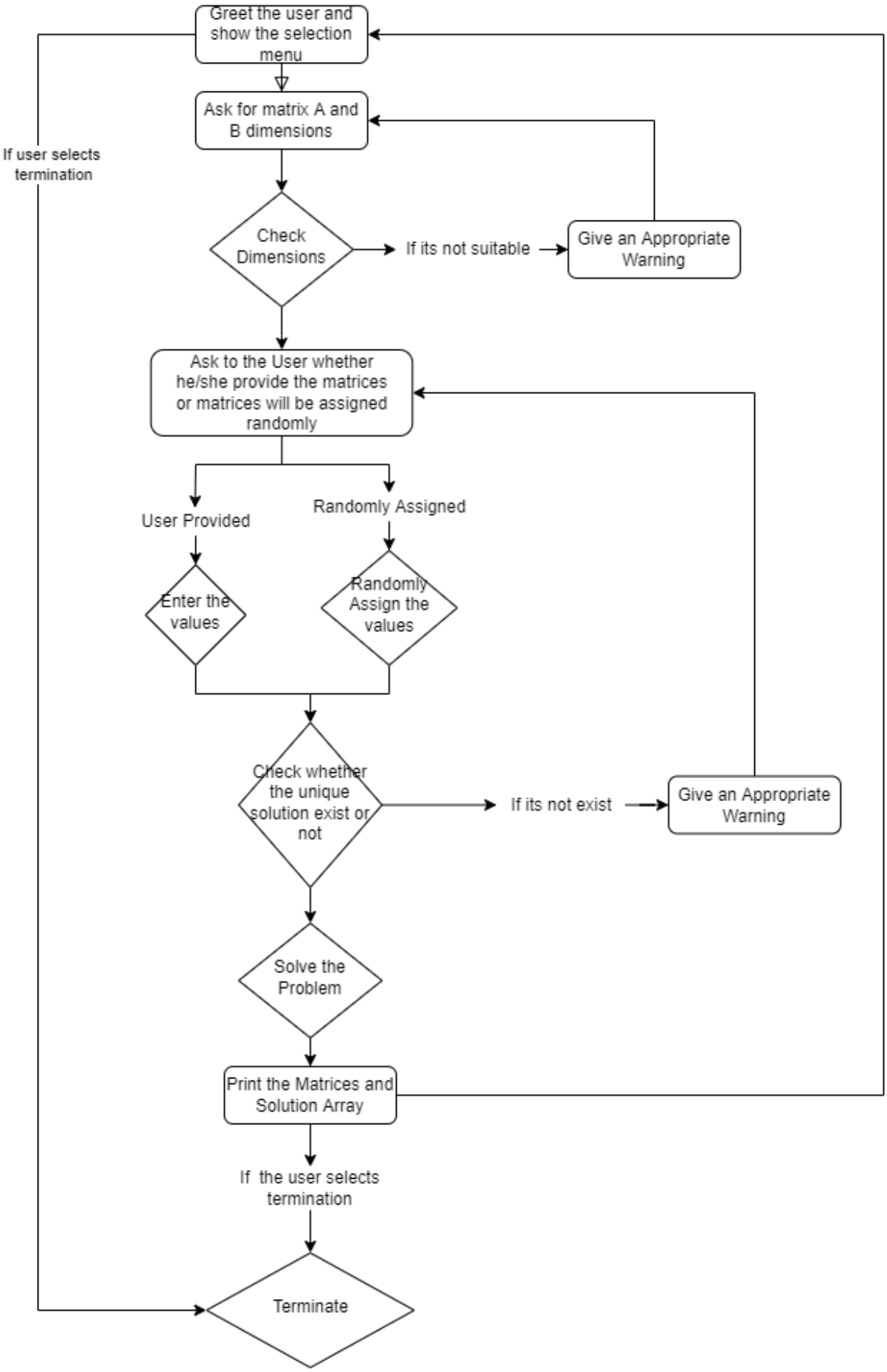
1. Program should greet the user with an appropriate menu
    a. In this menu, there will be two options which are
        i. "Equation Solver"
        ii. "Exit"
    b. If the "Exit" options selected by the user program will be closed immediately
    c. In case the user enters an input option other than the provided input option, the program should ask the user for an appropriate input option.
2. After selecting the "Equation Solver" option, ask the user for matrix **A** and array **b** dimensions. Their dimensions have to be appropriate to each other.
    a. If their dimensions are not appropriate then there should be a prompt that shows "Dimensions you have entered are not appropriate please re-enter the Dimensions"
    b. The user should re-enter dimensions
        i. If the user enters a floating number as a dimension the program should convert it to an integer and do the checking process.
3. There will be another menu which will be consist of
    a. User enters the values
        i. If that is selected user enters the matrixes values by hand
    b. Values will be randomly assigned
        i. The values of the matrix and arrays will be assigned randomly (please look at random number generation in the class notes)

c. In case the user enters an input option other than the provided input option, the program should ask the user for an appropriate input option.

4. Check whether the unique solution exists
    a. Tip: If any of the diagonal elements of **A** is 0 which means the **A** matrix is **singular** and a unique solution does not exist.
    b. If the unique solution does not exist your program should return to step 3 and re-initialize the matrix either by user provide or randomly.
5. Invert the Matrix **A** and multiply with array **b** and store the solution by a double array which will be named solution. (It's easy to invert a diagonal matrix)
6. Print the Matrix **A** array **b** and solution array
7. After the printing solution your program should **return to the menu from step 1**

Important Notes

- **All the external libraries and headers except the original C headers are not allowed to use.**
- **Robustness is important so your program should not just work for only one given dimension, it must work as long the user desires.**
- **Global variables, goto command, structs, and unions are not allowed to use.**
- **You have to create and use functions. The main block is just for definitions and program flow.**
- **You should use pointers for matrix A, array b, and solution array S.**
- **You should use call-by-reference and pointers as return variables for your functions (for your calculation functions).**
- **The assignment must be completely your own team's effort any kind of cheating (code copying etc.) will be punished severely.**
- **You can send an e-mail to altinisik19@itu.edu.tr for any kind of question about this project.**


Greet the user and show the selection menu
Ask for matrix A and B dimensions
If user selects termination
Check Dimensions
If its not suitable
Give an Appropriate Warning
Ask to the User whether he/she provide the matrices or matrices will be assigned randomly
User Provided
Randomly Assigned
Enter the values
Randomly Assign the values
Check whether the unique solution exist or not
If its not exist
Give an Appropriate Warning
Solve the Problem
Print the Matrices and Solution Array
If the user selects termination
Terminate