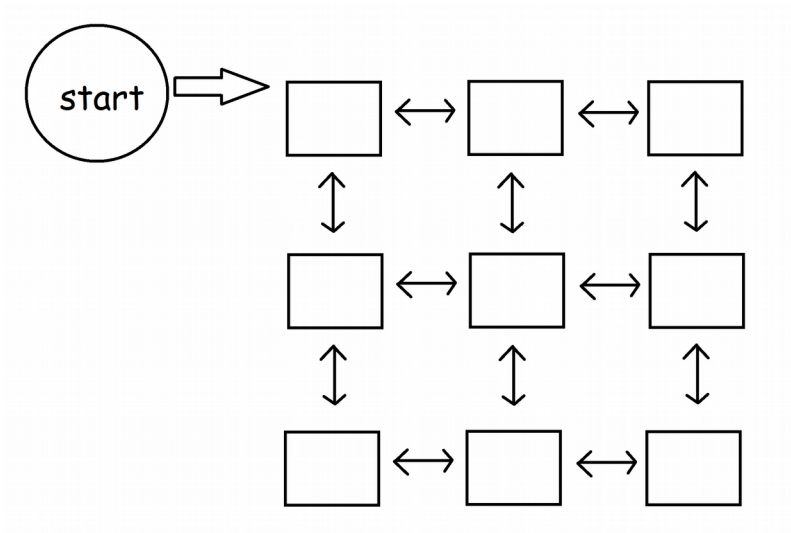


CS 302
Assignment 1
The Matrix Reloaded

In this assignment you need to write and execute a program that will perform addition, subtraction and multiplication of two valid matrices.

The matrices will be stored as double linked lists. Each item in the matrix will contain pointers to its neighboring values. Meaning the corners will have two pointers and the middle items will have four pointers. The entire matrix will only be accessible with a start pointer that points to the top right position as shown below:



To perform the addition and subtraction you must use a recursive function. You are not allowed to use a loop. You can however, use a loop for the matrix multiplication.

Your link list can be of integer or double type. Your choice, unless you want to use templates. That's fine too. I highly recommend your link list works before moving on to doing your matrix class.

Input will be given in the form of two matrices where the first two values of the input file will be the dimensions of the first matrix. Then the items of the first matrix, then the next two values will be the dimensions of the second matrix and then the items of the matrix. You must check if the matrix dimensions are valid for each operation and only perform those operations which the dimensions are valid for.

Notes:

- Comment your source code appropriately.
- Make sure you name your program file in accordance with the syllabus stipulations
- Test your program by running it a few times with different input to make sure the output is correct.

Test and execute your program by using data-in by Linux input redirection. If your executable code is `Ast#` and your data file is named `data#` execute as: `Ast# < data#`

Sample Input:

2 3
1 2 3
0 1 2

3 1
4
5
6

Sample output:

2 1
32
17

Part II:

When completed, create and submit a write-up (PDF Format) of no more than two pages (1 page is fine, half a page is not):

- Name, Assignment, Section
- Summary of the implemented linked list and matrix data structure
- A discussion on the efficiency of your matrix multiplication algorithm. Is there a way another way to write the algorithm to make the time complexity better? Are there any computations that are done more than once and that could be avoided by storing the results in a two-dimensional array?