

# 0301-346: Project #2

## Matrix Math

Fall 2020

### I. Introduction

For this project you will be exploring the implementation of matrix math using C++. There are third party libraries that provide matrix math operations, but for this project you will be implementing your own class object and overloading some C++ operators to achieve various matrix mathematical operations.

### II. Background

The goal of this project is to implement various matrix operations using a custom Matrix class. Therefore, it is important to establish how these operations are to be performed. All addition and subtraction operations are to be performed using element-wise addition/subtraction.

Consider the addition of two matrices **A** and **B** where the result is **C**. Both **A** and **B** are  $n \times m$  matrices.

$$\mathbf{A} + \mathbf{B} = \mathbf{C}$$

Element-wise addition will thus yield:

$$c_{ij} = a_{ij} + b_{ij}$$

where  $i = 0$  to  $n-1$  and  $j = 0$  to  $m-1$   
and  
 $n$  = number of rows,  $m$  = number of columns

Multiplication will be overloaded in this project and is to be implemented as matrix multiplication.

Consider the multiplication of matrix **A** and **B** (in that order) where the result is **C**.

$$\mathbf{A} * \mathbf{B} = \mathbf{C}$$

**A** is an  $n \times m$  matrix  
**B** is an  $m \times p$  matrix

A generic equation for performing matrix multiplication is:

$$c_{ij} = \sum_{k=0}^{m-1} a_{ik} b_{kj}$$

$$i = 0 \text{ to } n-1$$
$$j = 0 \text{ to } p-1$$

Note: Inner dimensions of the Matrix multiple MUST be equal.

### III. Project Requirements / Scoring

- A. [ 5pts] Class Requirements
  - i. Class **MUST** be called Matrix
  - ii. **MUST** store 2D Matrix data using a **private** member of type **double\*\***
  - iii. **MUST** use additional private member variables rows and cols
  - iv. **MUST** implement a destructor that deallocates all data properly
- B. [ 10pts] Constructors
  - i. Do NOT implement a default constructor
  - i. **MUST** implement Matrix(int,int) takes rows and cols
  - ii. **MUST** overload the Copy constructor
- C. [65 pts] Overloaded Operators
  - i. [10 pts] Stream insertion operator <<  
This will output matrix data with space separated column data and newline at the end of each row.
  - ii. [10 pts] Stream extraction operator >>  
  
First create a Matrix and define the row, cols. Then you can use 'cin' to input the values into the matrix. Any white space is valid for separating the input values.
  - iii. [10pts] Assignment operator (=)
  - iv. [ 5pts] Addition operator (+)
  - v. [ 5 pts] Subtraction operator (-)
  - vi. [ 5 pts] Addition assignment operator (+=)
  - vii. [ 5 pts] Subtraction assignment operator (-=)
  - viii. [15 pts] Multiplication (\*) as matrix multiplication
- D. [ 5pts] Asserting when operation cannot be properly performed.
  - i. Assert when (+, -, +=, -=) if the matrices are not the same size (row, cols)
  - ii. Asset when inner dimensions do not match for matric multiplication.
- E. [10pts] Program Execution
  - i. You will be provided a sample main.cpp and sample input script
  - ii. You must make sure your class can handle the main function and input values.
- F. [ 5pts] Peer Review
  - i. You will be required to fill out a survey on your teammates that will determine their score on this section of the project. It is to ensure participation from all members