

User Guide

Andrew Ye and James Ross

April 2024

Contents

| | | |
|---|-----------------------|---|
| 1 | Introduction | 3 |
| 2 | Accessing the Project | 3 |
| 3 | Class Methods | 3 |
| 4 | .rref() | 3 |
| 5 | .inverse() | 3 |
| 6 | .fraction() | 3 |
| 7 | + | 4 |
| 8 | * | 4 |

1 Introduction

This project was to program a way to row reduce the matrix, find the inverse of a matrix, multiply matrices, and add matrices.

2 Accessing the Project

Navigate to RREF repository. Download the file *all_together.py* and *execute.py*. Create an instance of the matrix class by assigning a variable to *matrix*(your matrix goes here) inside the *execute.py* file. Now, you can apply methods to this instance.

3 Class Methods

We created a matrix class in order to implement our goal.

| Operation | Function Name |
|------------------------------------|---------------|
| Row Reduced Echelon Form | .rref() |
| Inverse Matrix | .inverse() |
| Represent each entry as a fraction | .fraction() |
| Multiply Matrices | * |
| Add Matrices | + |

Note that each method returns an instance of the matrix class, so to access the matrix itself you will have to add *.matrix* to the end of the instance. Also, the *.fraction* method returns a list of list of strings, so *.fraction()* should be the last method applied to whatever operation you are applying to the matrix.

4 .rref()

This row reduces the matrix to echelon form.

5 .inverse()

This method returns the matrix B such that $rref(A) = BA$ where A is the input matrix. Thus when A is square and invertible, it will return the inverse of A where $I = rref(A) = BA$.

6 .fraction()

Returns the matrix as another matrix except each entry is a string in fraction form.

7 +

Matrix addition between two instances of matrix class.

8 *

Matrix multiplication between two instances of matrix class.