MATHS INNOVATIVE PROJECT REPORT

By:Krishang Gupta

DEVISING A PYTHON BASED MATHEMATICAL CALCULATOR TO DEAL WITH OPERATIONS ON MATRICES AND DETERMINANT

A PROJECT REPORT

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE

OF

BACHELOR OF TECHNOLOGY

IN

[MECHANICAL ENGINEERING]

SUBMITTED BY:

KRISHANG GUPTA

2K20/A9/49

UNDER THE SUPERVISION OF

DR. NILAM

MECHANICAL ENGINEERING

DELHI TECHNOLOGICAL UNIVERSITY

(FORMERLY Delhi College of Engineering)

Bawana Road, Delhi-110042

MARCH 2021

MECHANICAL ENGINEERING

DELHI TECHNOLOGICAL UNIVERSITY

(FORMERLY Delhi College of Engineering)

Bawana Road, Delhi-110042

CANDIDATE'S DECLARATION

I, (**Krishang Gupta, 2K20/A9/49**) student of B. Tech. (Mechanical Engineering) hereby declare that the project titled "**Devising A Python Based Mathematical Calculator To Deal With Operations On Matrices And Determinant** " submitted by me to the Department of Mechanical Engineering, Delhi Technological University, Delhi in partial fulfilment of the requirement for the award of the degree of Bachelor of Technology, is an original piece of work and is not copied from any source.This work has not previously formed the basis for the award of any Degree, Diploma Associateship, Fellowship or other similar title or recognition.

Place: Delhi   Krishang Gupta(2K20/A9/49)

Date:

MECHANICAL ENGINEERING

DELHI TECHNOLOGICAL UNIVERSITY

(FORMERLY Delhi College of Engineering)

Bawana Road, Delhi-110042

CERTIFICATE

I hereby certify that the project titled "**Devising A Python Based Mathematical Calculator To Deal With Operations On Matrices And Determinant**" which is submitted by Krishang Gupta (2K20/A9/49) [Mechanical Engineering], Delhi Technological University, Delhi in partial fulfilment of the requirement for the award of the degree of the Bachelor of Technology, is a record of the project work carried out by the student under my supervision. To the best of my knowledge this work has not been submitted in part or full for any Degree or Diploma to this University or elsewhere.

Place: Delhi **Dr. Nilam**

(Assistant Professor)

Date: **SUPERVISOR**

MECHANICAL ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITY
(FORMERLY Delhi College of Engineering)
Bawana Road, Delhi-110042

ABSTRACT

This **GUI based offline Calculator** is specially designed for certain basic operations on Matrices and Determinants with the functionality to deal with Matrix Addition and Subtraction,Matrix Multiplication,Transpose,Adjoint and Inverse calculations and finding out the Determinant of any Matrix.

Currently,the Program deals only with the Square Matrices of 2^{nd} and 3^{rd} order. However,other features like **User-defined order of the Matrix,Echelon form of the selected Matrix and Calculation of the Rank of the chosen Matrix** would be added in the Program in the fullness of time.

The Calculator is programmed using the Tkinter Module of the Python Programming Language, aided with some pre-programmed Python libraries like the 'Numpy'library.The code is written in the Jupyter notebook and the Tkinter module is utilised to create the Graphical User Interface for the Calculator.

There are two sections in the Calculator,one each for 2x2 and 3x3 Square Matrix.A scrollbar is attached with the Calculator for the user to switch between the two sections easily.

The user needs to select any one type of the Matrix followed by the desired operation he wants the Calculator to do.Only the section corresponding to the order of the selected Matrix would get activated and take number inputs by the user.The other section would get disabled temporarily.

To get the Output,the User is required to click the ‘CALCULATE’ button situated at the centre and the Calculator can be closed anytime by making use of the ‘EXIT’button.

MECHANICAL ENGINEERING

DELHI TECHNOLOGICAL UNIVERSITY

(FORMERLY Delhi College of Engineering)

Bawana Road, Delhi-110042

ACKNOWLEDGEMENT

In performing my Innovative Project, I took the guidance of some respected teachers, who deserve my greatest gratitude. The completion of this Project gives me much pleasure. I would like to show my gratitude towards **Dr.Nilam Ma'am**, Mentor for Mathematics project. I would also like to extend my deepest gratitude to,**Mr.Ankit Sharma Sir** for guiding me in the completion of this Project.

Many people, especially **Mr. Ankit Mittal Sir (**FIITJEE, South Delhi**)** and **Mrs Kanchan Mishra Ma'am**(Director,Kanchan Coaching Centre) have made valuable suggestions on this Project which gave me an inspiration to improve my Project. I thank all the people for their help directly or indirectly to complete my Project.

In addition, I would like to thank Department of Mechanical Engineering, Delhi Technological University for giving me the opportunity to work on this Project.

Table of Contents

CHAPTER 1

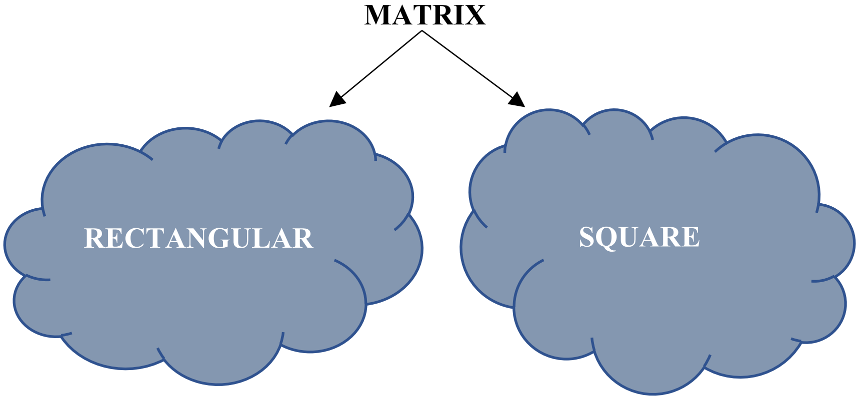
INTRODUCTION

1.1 History of Matrices

- The origin of mathematical matrices lies with the study of systems of simultaneous linear equations. The theory of matrices was developed by a mathematician named **Gottfried Leibniz.**
- The term *matrix* was introduced by the 19th-century English mathematician **James Sylvester,** but it was his friend the mathematician **Arthur Cayley** who developed the algebraic aspect of matrices in two papers
- **Cayley** tried to apply them to the study of systems of linear equations, where they are still very useful. He recognized, certain sets of matrices form algebraic systems in which many of the ordinary laws of arithmetic (e.g., the associative and distributive laws) are valid but in which other laws (e.g., the commutative law) are not valid.

1.2 Matrix and its Types

- A Matrix is a set of numbers which when arranged in rows and columns forms a rectangular array. The numbers are known as elements or entries of the matrix.



- If there are m rows and n columns, the matrix is said to be an "m by n" matrix, written as "$m \times n$." Example:

$$\begin{bmatrix} 1 & 3 & 8 \\ 2 & -4 & 5 \end{bmatrix}$$

m=2 and n=3

The given Matrix is a **2X3 Matrix**

Such a Matrix (m and n are unique) is known is Rectangular Matrix.

- If there are n rows and n columns, the matrix is said to be an "n by n" matrix, written as "$n \times n$." Example:

$$\begin{bmatrix} 2 & 3 \\ 3 & 4 \end{bmatrix}$$

m=2 and n=2

The given Matrix is a **2X3 Matrix**

Such a Matrix (m and n are identical i.e m=n) is known as Square Matrix.

1.2.1 Types of Rectangular Matrices

1)Row Matrix

A matrix having only one row but any number of columns is called a **row matrix**.

$$A = [1\ 2\ 4\ 5]$$

2)Column Matrix

A matrix having only one column but any number of rows is called a **column matrix**.

$$A = \begin{bmatrix} 1 \\ 2 \\ 4 \\ 5 \end{bmatrix}$$

3)Null Matrix

If in a matrix all the elements are zero then it is called a **zero matrix** and it is generally denoted by 0

$$A = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

Zero matrix

4)Horizontal Matrix

A matrix of order m x n is a **horizontal matrix** if n > m

$$\begin{bmatrix} 3 & 2 & 11 & 5 \\ 2 & 9 & -1 & 6 \end{bmatrix}$$

5)Vertical Matrix

A matrix of order m x n is a **horizontal matrix** if m > n

$$\begin{bmatrix} 11 & 5 \\ -1 & 6 \\ 0 & 7 \\ 7 & 9 \end{bmatrix}$$

1.2.2 Types of Square Matrices

1)Identity Matrix

An **Identity Matrix** has **1**s on the main diagonal and **0**s everywhere else:

$$I = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

A 3×3 Identity Matrix

- It is square (same number of rows as columns)
- It can be large or small (2×2, 100×100, ... whatever)
- Its symbol is the capital letter **I**

It is the matrix equivalent of the number "1", when we multiply with it the original is unchanged:

$A \times I = A$

$I \times A = A$

2)Diagonal Matrix

A diagonal matrix has zero anywhere not on the main diagonal:

$$\begin{bmatrix} 2 & 0 & 0 \\ 0 & 8 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

A diagonal matrix

3)Scalar Matrix

A scalar matrix has all main diagonal entries the same, with zero everywhere else:

$$\begin{bmatrix} 5 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 5 \end{bmatrix}$$

A scalar matrix

4)Triangular Matrix

Lower triangular is when all entries above the main diagonal are zero:

$$\begin{bmatrix} 5 & 0 & 0 \\ 2 & 1 & 0 \\ 7 & 6 & -3 \end{bmatrix}$$

A lower triangular matrix

Upper triangular is when all entries below the main diagonal are zero:

$$\begin{bmatrix} 2 & -2 & 7 \\ 0 & 4 & 11 \\ 0 & 0 & 5 \end{bmatrix}$$

An upper triangular matrix

5)Zero Matrix (Null Matrix)

All the entries are 0:

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Zero matrix

6)Symmetric

In a Symmetric matrix matching entries either side of the main diagonal are **equal**, like this:

$$\begin{bmatrix} 3 & 2 & 11 & 5 \\ 2 & 9 & -1 & 6 \\ 11 & -1 & 0 & 7 \\ 5 & 6 & 7 & 9 \end{bmatrix}$$

Symmetric matrix

It must be square, and is equal to its own transpose

$A = A^T$

7)Hermitian

A Hermitian matrix is symmetric except for the ***imaginary parts*** that swap sign across the main diagonal:

$$\begin{bmatrix} 3 & 2+3i & -2i & 5-i \\ 2-3i & 9 & 12 & 1+4i \\ 2i & 12 & 1 & 7 \\ 5+i & 1-4i & 7 & 12 \end{bmatrix}$$

Hermitian matrix

Here **+i** changes to **−i** and vice versa.(Changing the sign of the second part is called the conjugate)

A Hermitian matrix is equal to its own **conjugate transpose**:

$A = A^T$

This also means the main diagonal entries must be purely real (to be their own conjugate).

It is named after French mathematician Charles Hermite.

1.3 Determinant of a Matrix $|A|$

- Determinant of a Matrix is a special number,defined only for square matrices.
- Determinant is used at many places in calculus and other matrix related algebra.It actually represents the matrix in term of a real number which can be used in solving system of linear equation and finding the inverse of a matrix.

1.3.1 2x2 Matrix

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

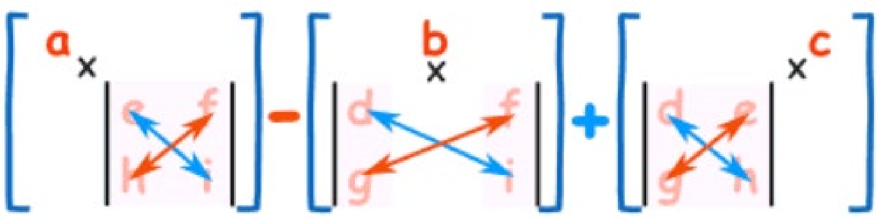
The determinant is:

$$\boxed{|A| = ad - bc}$$

1.3.2 3x3 Matrix

$$A = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}$$

The determinant is:



$$\boxed{|A| = a(ei - fh) - b(di - fg) + c(dh - eg)}$$

1.3.3 Singular and Non Singular Matrix

Singular Matrices are those Matrices whose value of determinant is 0

The value of Determinant of a Non Singular Matrix is anything except 0

$$\begin{bmatrix} 2 & 4 & 6 \\ 2 & 0 & 2 \\ 6 & 8 & 14 \end{bmatrix}$$

The Determinant is given by-

2(0-16) – 4(28-12) + 6(16-0) = -2(16) + 2(16) = 0

As the Determinant is equal to 0,it is a Singular Matrix

$$\begin{bmatrix} 2 & 6 & 1 \\ -3 & 0 & 5 \\ 5 & 4 & -7 \end{bmatrix}$$

The Determinant is given by-

2(0-20) - 6(21-25) + 1(-12-0) = 2(-20) + 24 – 12 = -28

As the Determinant is not equal to 0,it is a Non Singular Matrix

1.4 Transpose of a Matrix $\mathbf{A}^{\mathrm{T}}$

The transpose of a Matrix is the new Matrix which is obtained by exchanging the rows and columns of that Matrix

$$[\mathbf{A}^{\mathrm{T}}]_{ij} = [\mathbf{A}]_{ji}$$

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}^{\mathrm{T}} = \begin{bmatrix} 1 & 3 & 5 \\ 2 & 4 & 6 \end{bmatrix}$$

1.5 Adjoint of a Matrix adj(A)

The adjoint of a Matrix A is the transpose of the Cofactor Matrix of A

$$A = \begin{bmatrix} 3 & 1 & -1 \\ 2 & -2 & 0 \\ 1 & 2 & -1 \end{bmatrix}$$

Cofactor of $3 = A_{11} = \begin{vmatrix} -2 & 0 \\ 2 & -1 \end{vmatrix} = 2$

Cofactor of $1 = A_{12} = -\begin{vmatrix} 2 & 0 \\ 1 & -1 \end{vmatrix} = 2$

Cofactor of $-1 = A_{13} = \begin{vmatrix} 2 & -2 \\ 1 & 2 \end{vmatrix} = 6$

Cofactor of $2 = A_{21} = -\begin{vmatrix} 1 & -1 \\ 2 & -1 \end{vmatrix} = -1$

Cofactor of $-2 = A_{22} = \begin{vmatrix} 3 & -1 \\ 1 & -1 \end{vmatrix} = -2$

Cofactor of $0 = A_{23} = -\begin{vmatrix} 3 & 1 \\ 1 & 2 \end{vmatrix} = -5$

Cofactor of $1 = A_{31} = \begin{vmatrix} 1 & -1 \\ -2 & 0 \end{vmatrix} = -2$

Cofactor of $2 = A_{32} = -\begin{vmatrix} 3 & -1 \\ 2 & 0 \end{vmatrix} = -2$

Cofactor of $-1 = A_{33} = \begin{vmatrix} 3 & 1 \\ 2 & -2 \end{vmatrix} = -8$

The cofactor matrix of A is $[A_{ij}] = \begin{bmatrix} 2 & 2 & 6 \\ -1 & -2 & -5 \\ -2 & -2 & -8 \end{bmatrix}$

$$\begin{aligned} adj\,A &= (A_{ij})^T \\ &= \begin{bmatrix} 2 & -1 & -2 \\ 2 & -2 & -2 \\ 6 & -5 & -8 \end{bmatrix} \end{aligned}$$

1.6 Inverse of a Matri $\mathbf{A}^{-1}$

- The Inverse of a Matrix is that Matrix which when multiplied with the original Matrix will give as an Identity Matrix.

$$\mathbf{AB} = \mathbf{BA} = \mathbf{I}_n$$

- Inverse of a Matrix is only defined if the Matrix is a Non-Singular Matrix i.e the Determinant of the Matrix is not equal to 0.

If A is an invertible matrix, then

$$A^{-1} = \frac{1}{\det(A)} \operatorname{adj}(A)$$

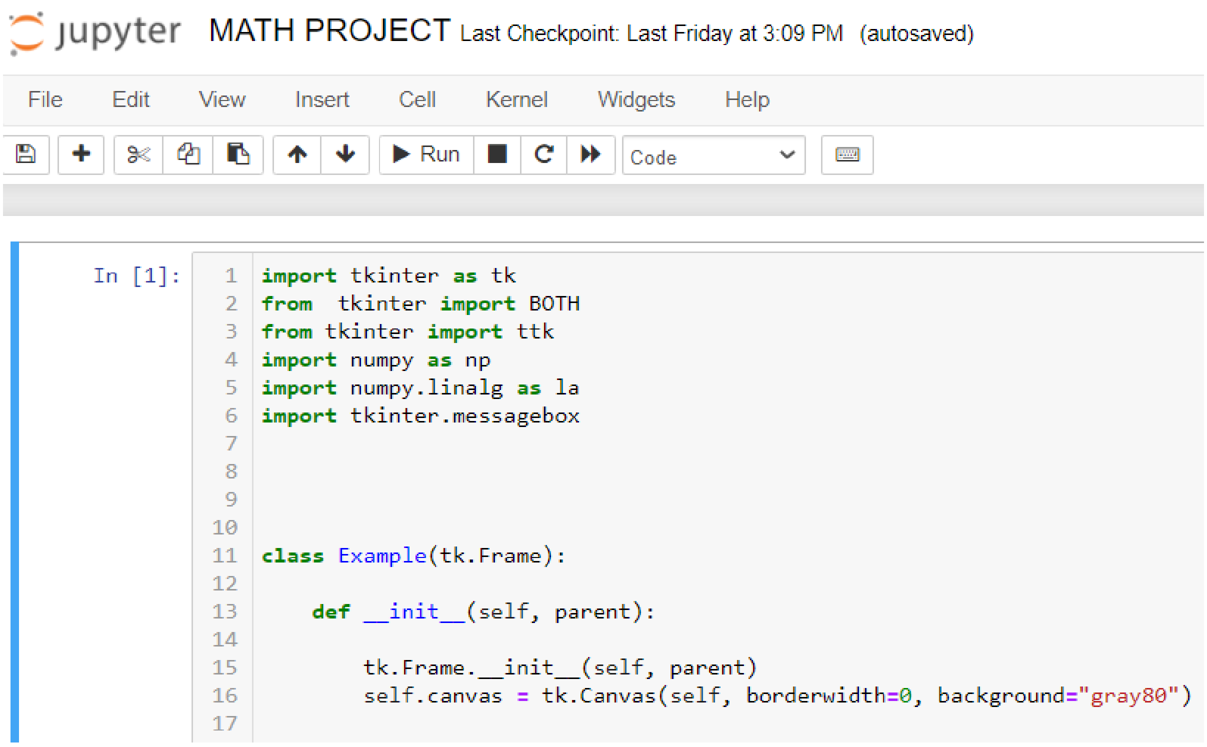
CHAPTER 2
METHODOLOGY

2.1 Generation and optimization of the Python code in Jupyter Notebook using the Tkinter Library of the Python Module

- A Graphical User Interface(GUI) based Python Code was created for the Matrix Calcualtor using the Tkinter Library.The Numpy Library of Python was harnessed for carrying out the operations on Matrices.
- The 2x2 and 3x3 Matrix selection options were added to the Program in the form of Radiobuttons and the user was required to select any of the Radiobutton as per his desire.
- All the 7 Major operations(**Add,Subtract,Multiply,Determinant,Adjoint,Transpose and Inverse**) were added next to the Matrix selection option.In case the user forgot to select the Operation from the Operations section,the Calculator showed an Error and requested the user to select the operation first and then run the Program.This was made possible by the Tkinter Messagebox utility.
- A separate section was created for the Determinant calcualtions for both 2md and 3rd order Matrices and the result of the same was displayed on the same window in an Entry Box.
- Two separate functions with the function names **op1()**(op1 stands for operations 1) and **op2()**(op2 stands operations 2) were coded to deal with the operations on the 2x2 and 3x3 Matrices respectively.
- A function named **changer()** was thought upon and coded to manage the Activation and Deactivation of the Entry Boxes based on the Matrix Selected and the respective Operation the user selected.

For example, only the 2x2 Determinant section was activated when the user selected **2x2 Matrix** and then the **Determinant** option.

- To improve the code, it was later modified for letting the user input integers as well as decimal digits in the Matrix Entry Boxes.A scrollbar widget was also added to the Tkinter window in order to surf from one section to another in the Calculator.



```
        def changer():

            l18.configure(text="")
            l18.update()

            l19.configure(text="")
            l19.update()

            l20.configure(text="")
            l20.update()



            if v.get()==0:#If user selected 2x2 matrix


                    l30.configure(text="")
                    l30.update()
                    l31.configure(text="")
                    l31.update()
```

```

            elif v.get()==1:#If user selected 3x3 matrix






                    for i in range(1,19):#close the entry boxes in 2x2 columns and close the buttons in 2x2
                            entry[i].configure(state="disabled")
                            entry[i].update()

                    for i in range(1,3):#close the buttons in 2x2
                            button[i].configure(state="disabled")
                            button[i].update()


                    for i in range(19,46):#open the entry boxes of 3x3 first row
                            entry[i].configure(state="normal")
                            entry[i].update()

```

```
                def op1():#all operations  2x2
                                for i in range(10,14):
                                            entry[i].configure(state="normal")

                                c=np.matrix([[0,0],[0,0]])
                                if o.get()==0:#add

                                    A=np.matrix([   [float(entry[1].get()),floa
                                    B=np.matrix([   [float(entry[6].get()),floa
                                    c=A+B
```

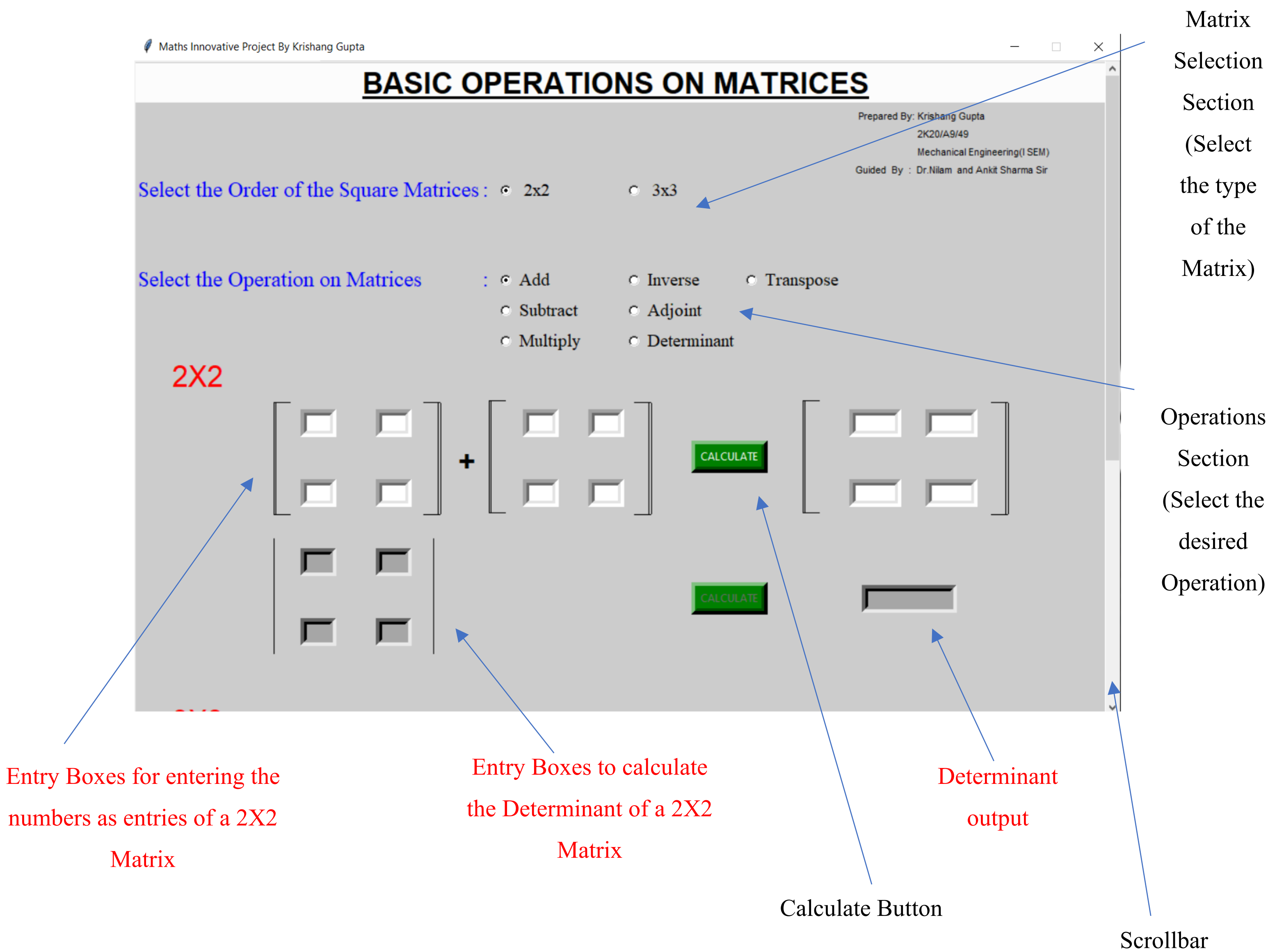
```
                def op2():#all operations 3x3
                                for i in range(37,46):
                                            entry[i].configure(state="normal")
                                c=np.matrix([[0,0,0],[0,0,0],[0,0,0]])
```

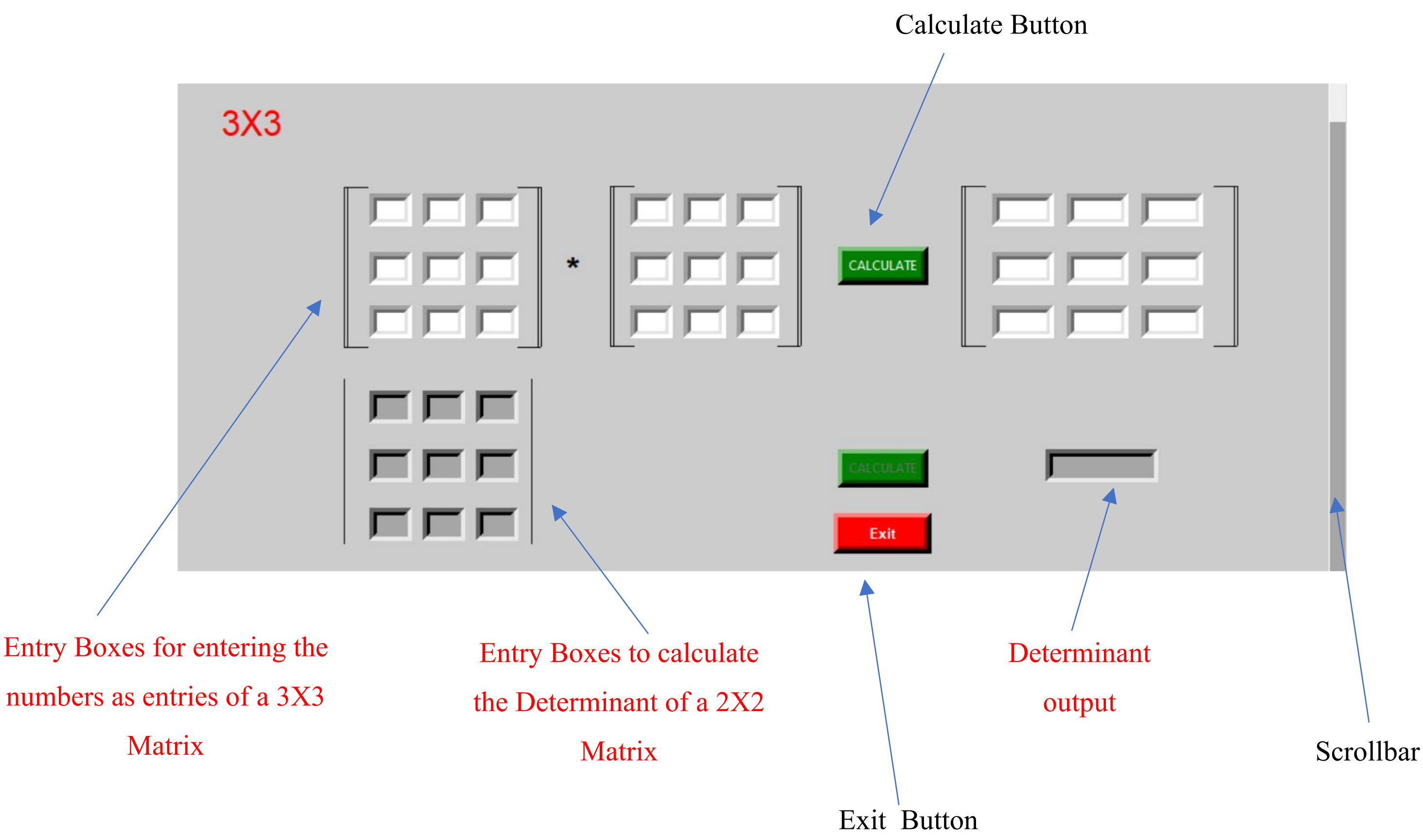
(Fig- Snippets of the code)

2.2 Testing the code for Errors

- The code was tested for multiple possible combinations and the results were verified manually to check the accuracy of the code.
- Different scenerios were taken care of at this stage of Programming.The possibilities of errors based on certain specific user selctions of the Operations on the Matrix feeded to the Calculator was removed. Cases such as **Displaying of an error message in case the user desires to calculate the inverse of a Singular Matrix** were considered and an appropriate code was implemented to cater the needs of each scenerio**.**

MAIN WINDOW OF THE CALCULATOR



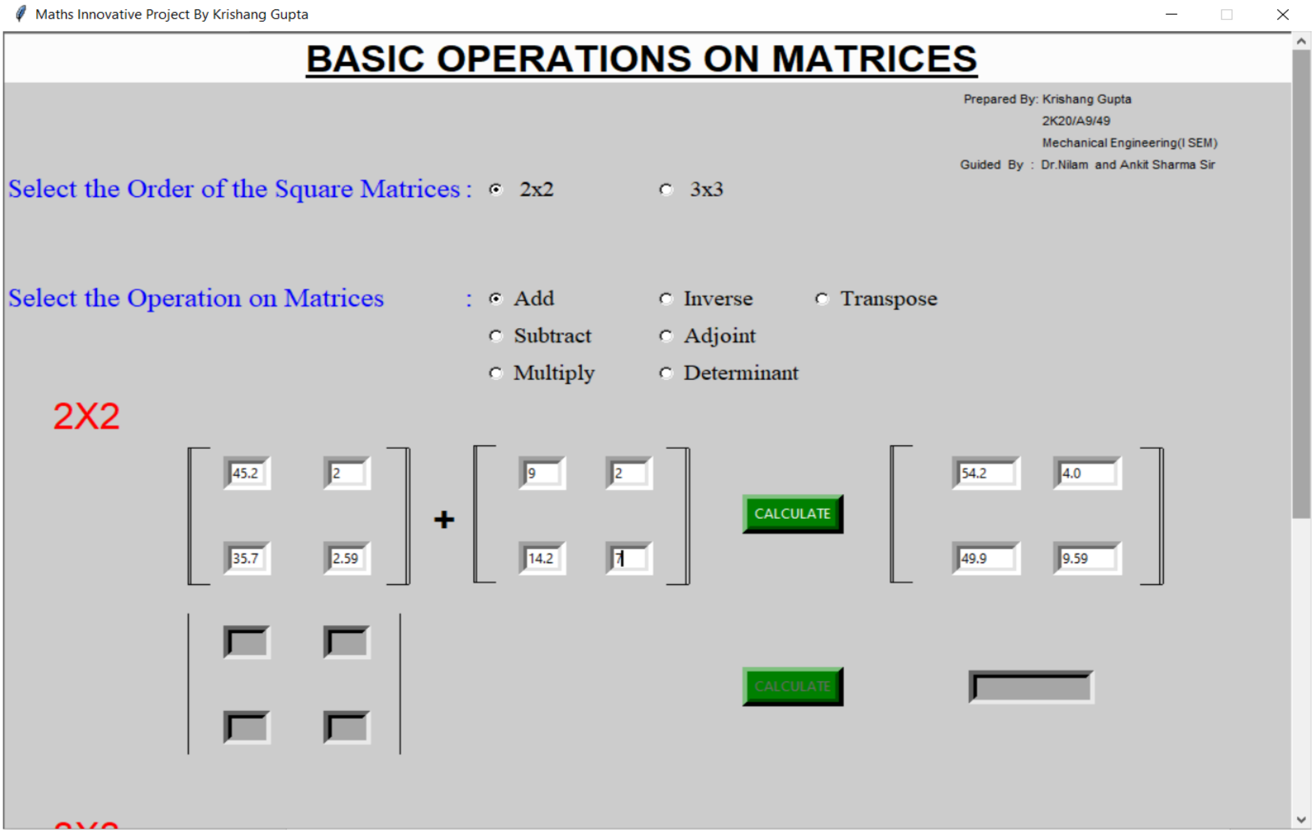

Calculate Button
3X3
*
CALCULATE
CALCULATE
Exit
Entry Boxes for entering the numbers as entries of a 3X3 Matrix
Entry Boxes to calculate the Determinant of a 2X2 Matrix
Determinant output
Scrollbar
Exit Button

CHAPTER 3

RESULTS AND CONCLUSION

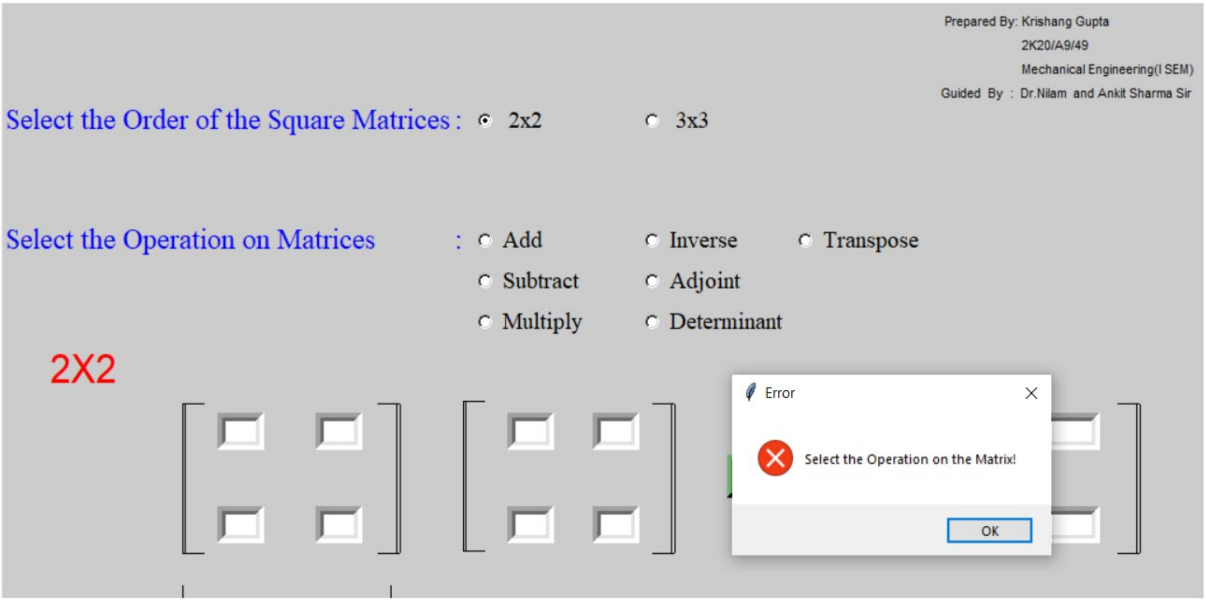
3.1 CASE-1

Matrix: 2x2 Operation: Add



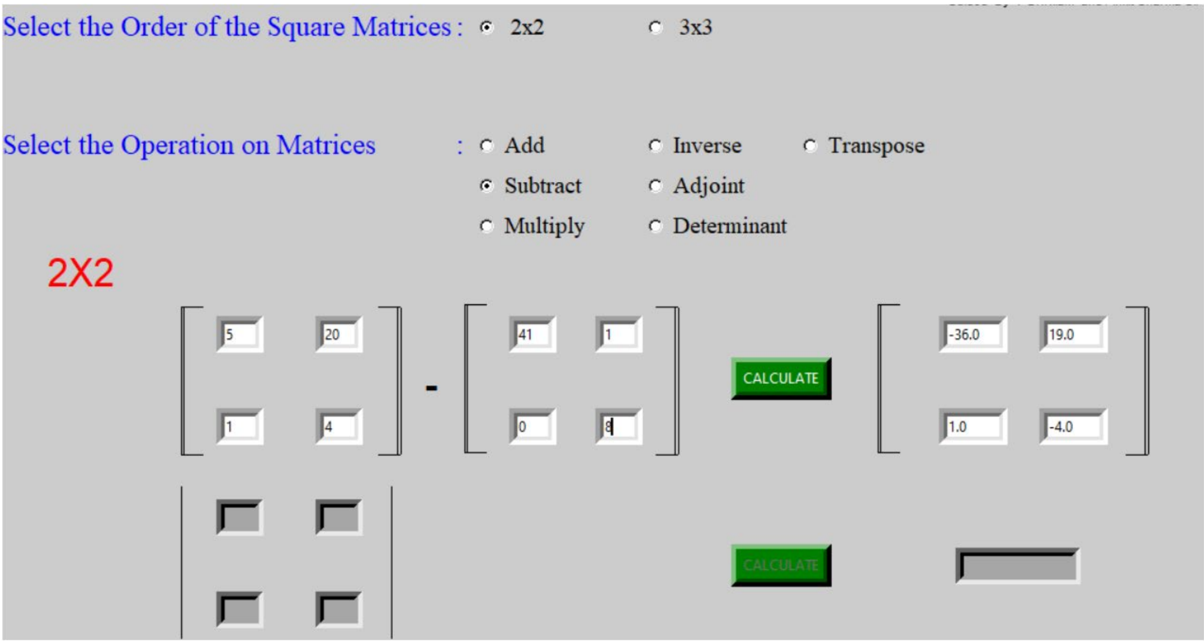
3.2 CASE-2

Matrix: 2x2 Operation: Not selected



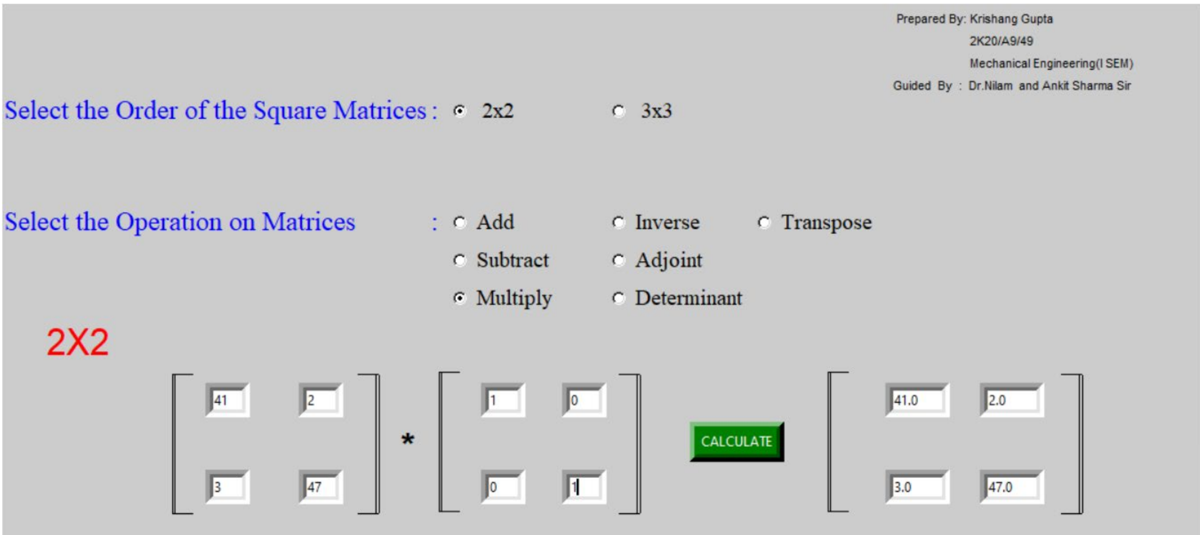
3.3 CASE-3

Matrix: 2x2   Operation: Subtract



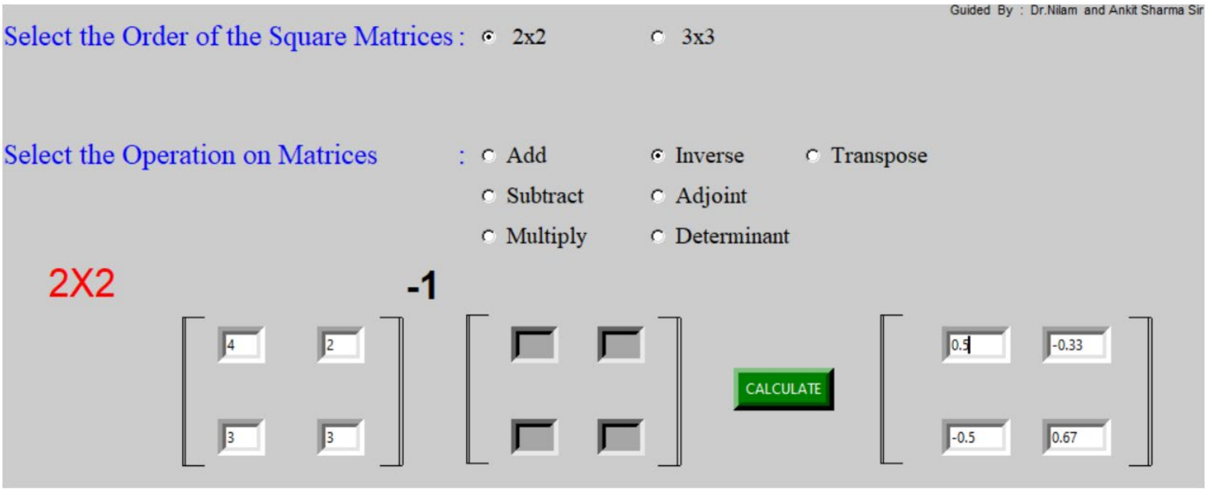
3.4 CASE-4

Matrix: 2x2   Operation: Multiply



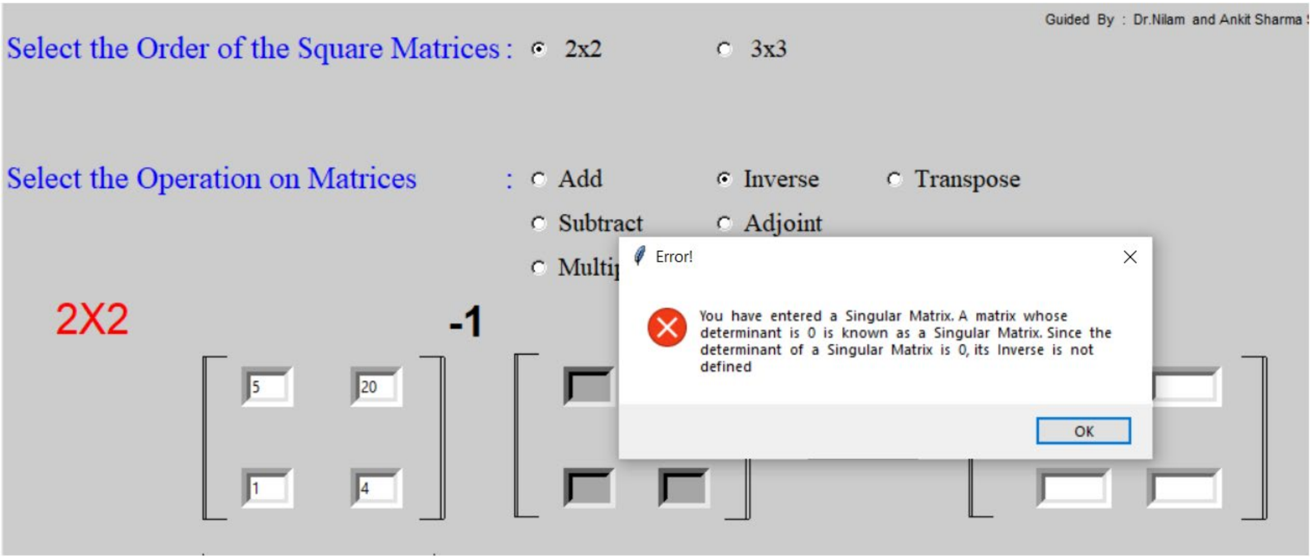
3.5 CASE-5

Matrix: 2x2   Operation: Inverse



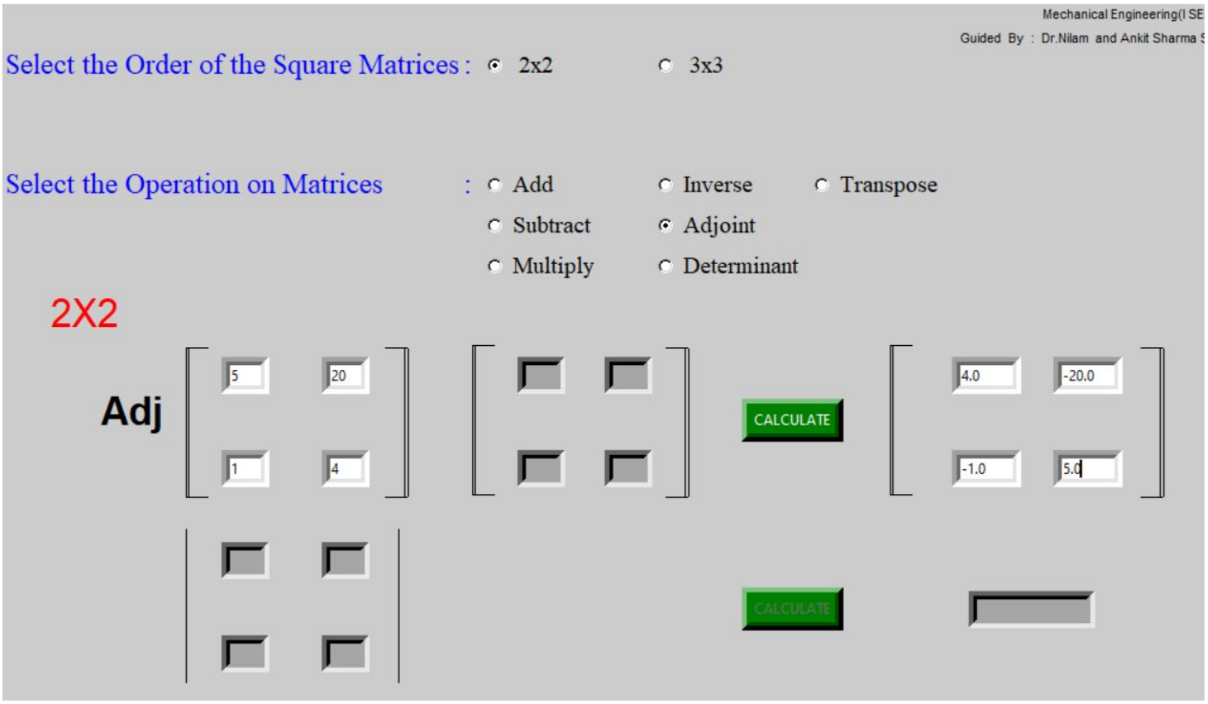
3.6 CASE-6

Matrix: 2x2   Operation: Inverse (Singular Matrix)



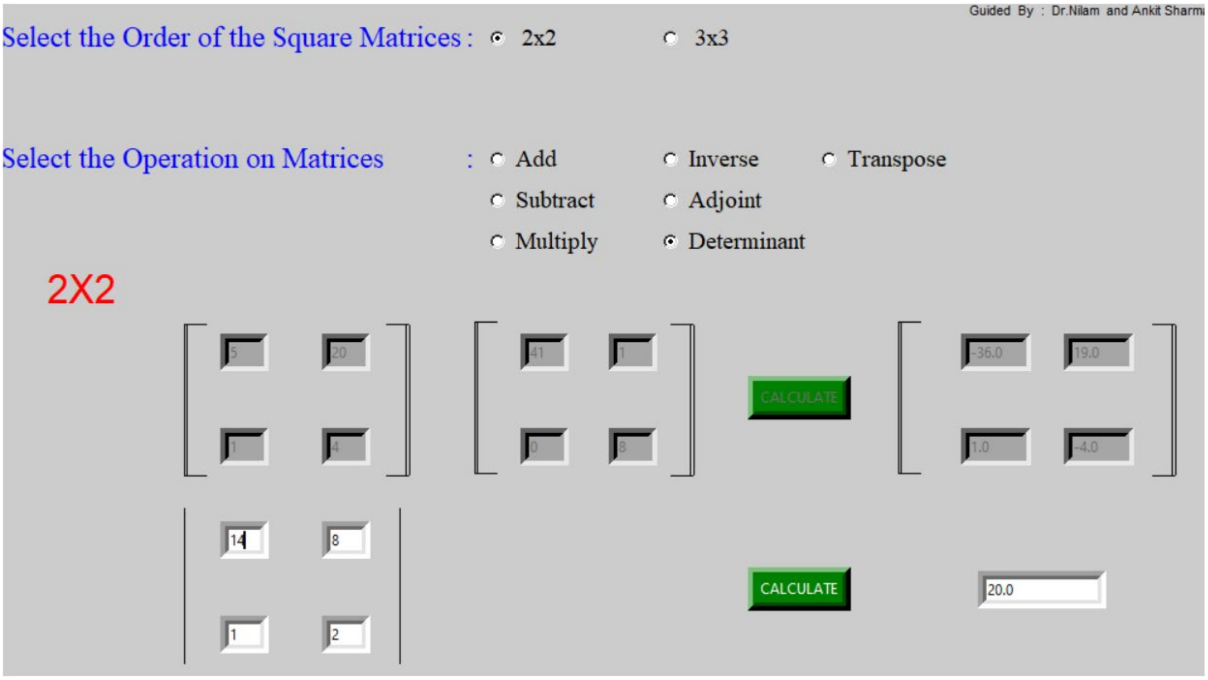
3.7 CASE-7

Matrix: 2x2 Operation: Adjoint



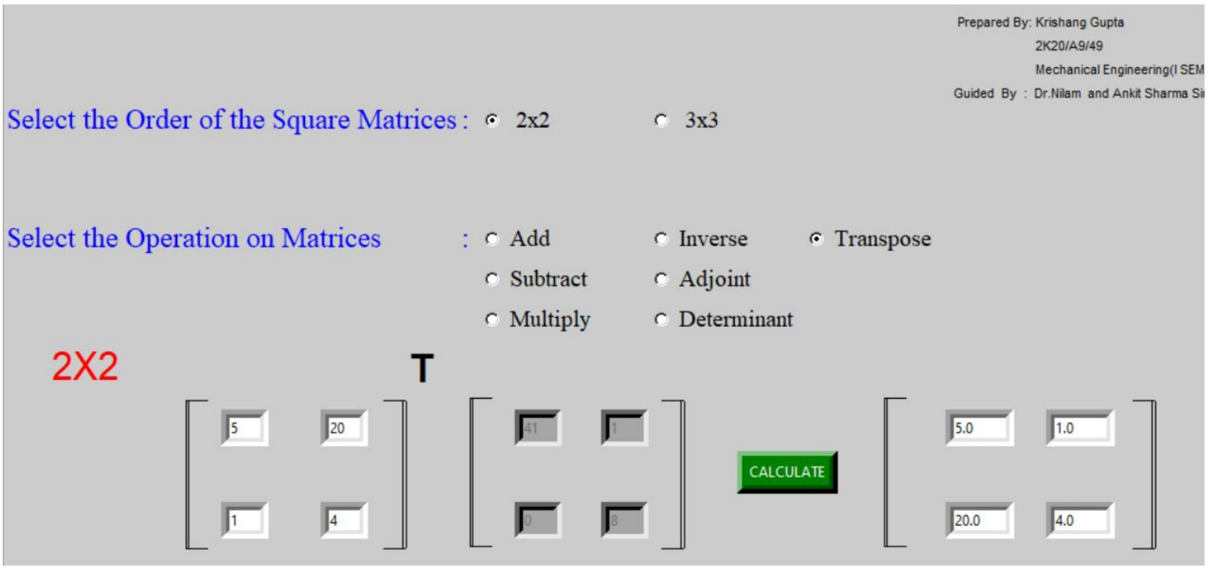
3.8 CASE-8

Matrix: 2x2 Operation: Determinant



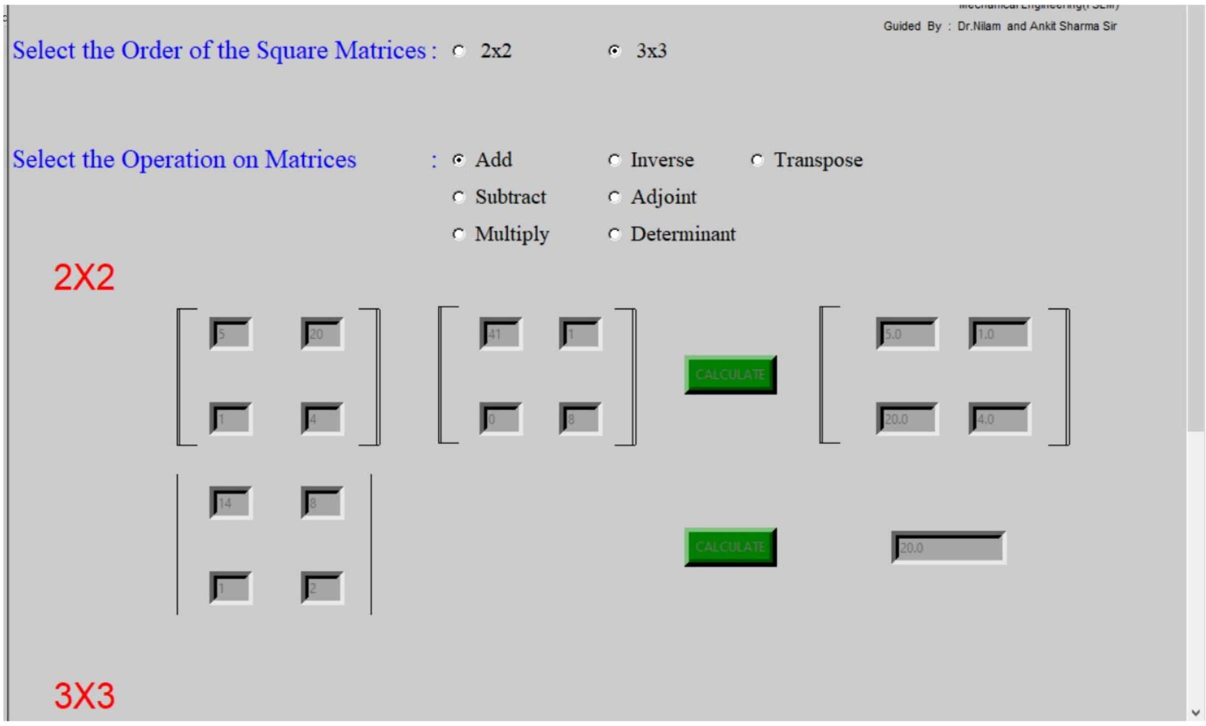
3.9 CASE-9

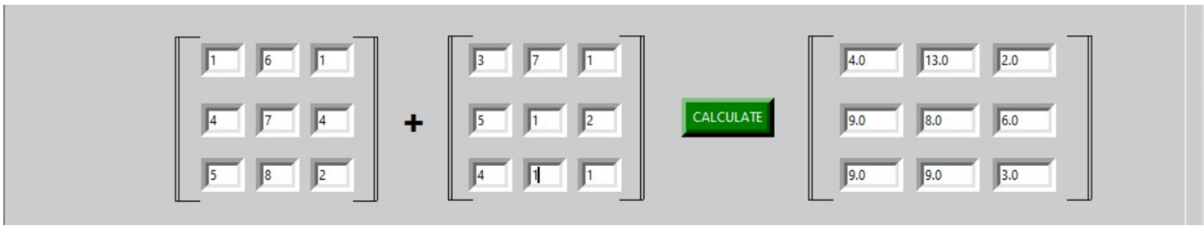
Matrix: 2x2   Operation: Transpose



3.10 CASE-10

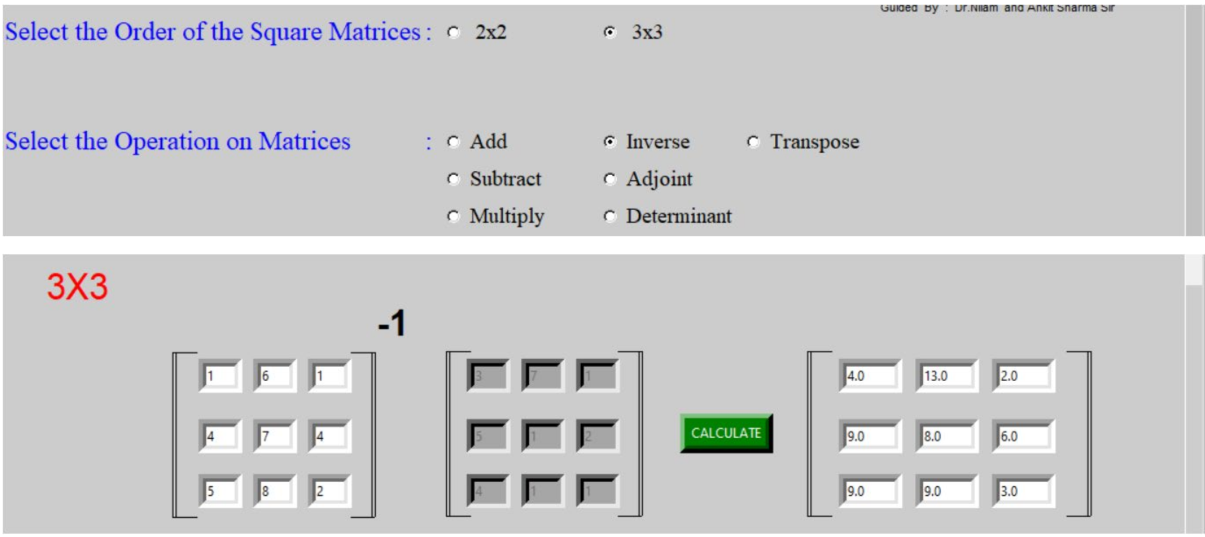
Matrix: 3x3 Operation: Add





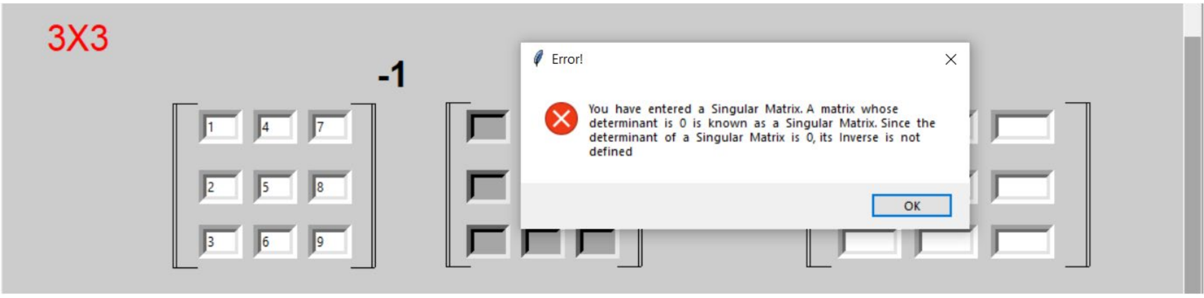
3.11 CASE-11

Matrix: 3x3 Operation: Inverse

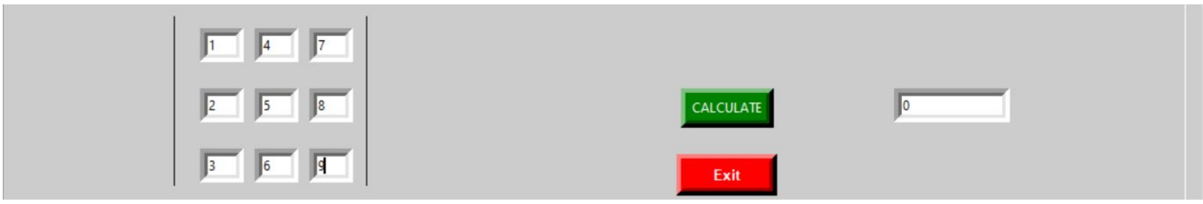


3.12 CASE-12

Matrix: 3x3 Operation: Inverse(Singular Matrix)

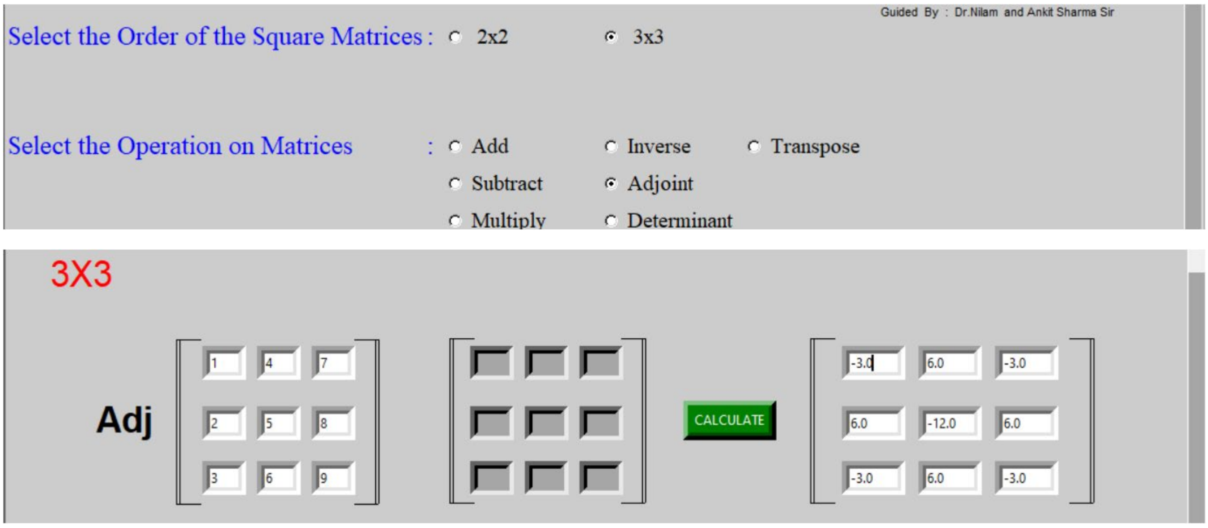


Checking the result:



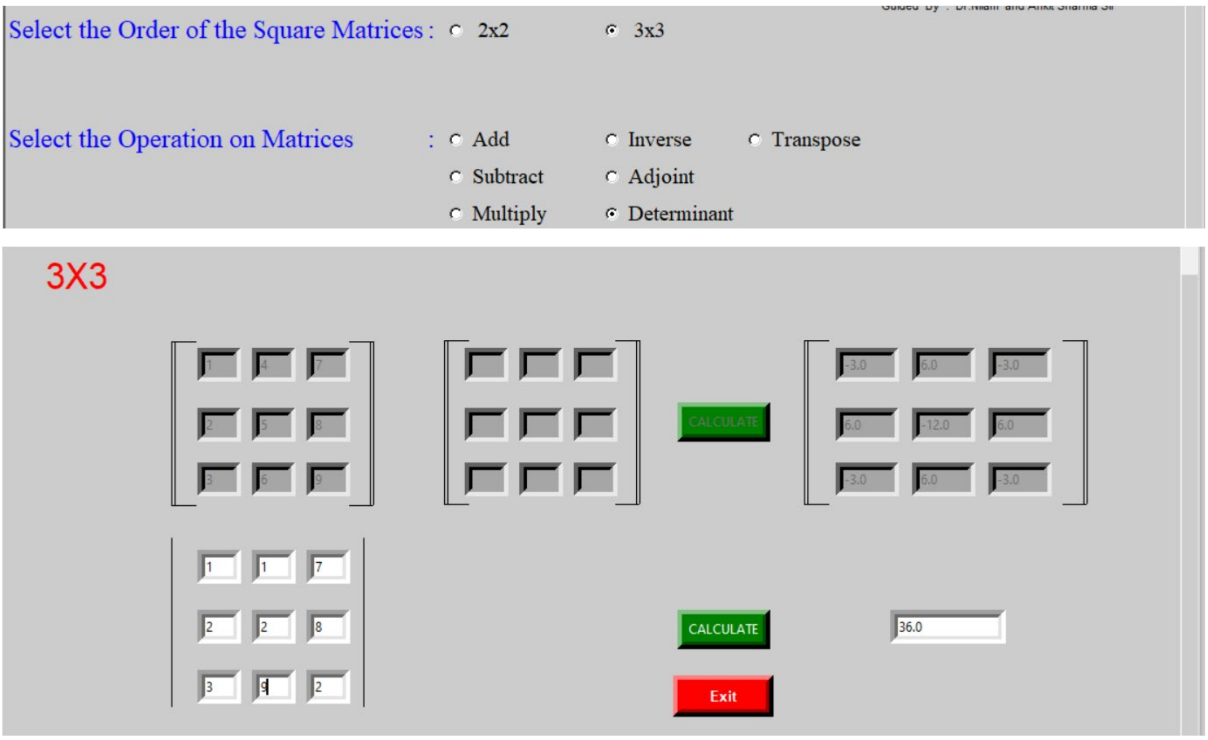
3.3 CASE-13

Matrix: 3x3 Operation: Adjoint



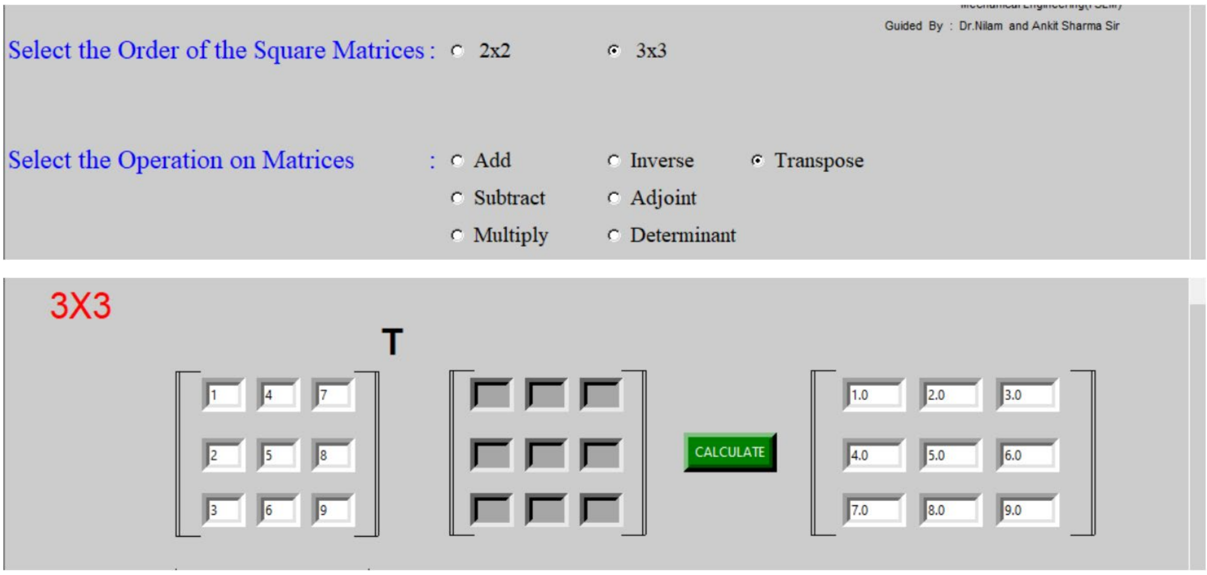
3.4 CASE-14

Matrix: 3x3 Operation: Determinant



3.5 CASE-15

Matrix: 3x3 Operation: Transpose



CHAPTER 4
FUTURE PROSPECTS

1)Modifying the Calculator to cater the calculations for Matrix of any order

2)Improving the GUI such that the user doesn't need to scroll down the Program to operate on 3X3 Matrices.

3)Incorporating the Rank Calculation Functionality and modifying the Calculator to calculate the Echelon Form of any Matrix

REFERENCES

1. https://www.geeksforgeeks.org/python-tkinter-tutorial/
2. https://www.tutorialspoint.com/python/python_gui_programming.htm
3. https://www.javatpoint.com/python-tkinter
4. https://www.geeksforgeeks.org/python-programming-language/
5. NCERT Class 12th