```
import numpy as np

def matrix_input():
    rows = int(input("Enter the number of rows: "))
    columns = int(input("Enter the number of columns: "))
    matrix = np.zeros((rows,columns))
    for i in range(0,rows):
        for j in range(0,columns):
            matrix[i,j] = int(input(f"Column: {j+1} Row: {i+1} ->"))
    return matrix

def menu():
  print("1. Matrix Addition\n2. Matrix Subtraction\n3. Scalar Matrix Multiplication\n4. Elementwise Matrix Multiplication\n5. Matrix Multiplication\n6. Matrix Transpose\n7. Trace of a Matrix\n8. Solve System of Linear Equations\n9. Determinant\n10.  Inverse\n11.  Singular Value Decomposition\n12.  Eigen Value\n13.  Search an Element\n14.  Difference of Sum of Upper and Lower Triangular Matrix\n15.  Exit")

def validation():
  print("Checking if the matrices are of same shape:")
  if matrix1.shape == matrix2.shape:
    print("Passed, you may continue")
  else:
    print("Falied")
    quit()



#Getting inputs from user
print("Enter the details of the 1st matrix:\n")
matrix1 = matrix_input()
print("Enter the details of the 2nd matrix:\n")
matrix2 = matrix_input()

print("The given matrices are")
print(matrix1)
print("\t")
print(matrix2)

#Menu
print("\nWelcome to the matrix operation calculator:\n")


condition = True

while condition:
  menu()
  option = int(input("Enter the option: "))

  if option == 1:
    validation()
    print("\nMatrix Addition:\n")
```

```
  print(f"The final matrix is:\n {matrix1 + matrix2}\n")

if option == 2:
  validation()
  print("\nMatrix Subtraction:\n")
  print(f"The final matrix is:\n {matrix1 - matrix2}\n")

if option == 3:
  print("\nScalar Matrix Multiplication:\n")
  a = int(input("Enter the scalar number"))
  choice = input("Choose between a)matrix1 and b)matrix2: ")
  if choice == 'a':
    print(f"The final matrix is:\n {matrix1 * a}\n")
  if choice == 'b':
    print(f"The final matrix is:\n {matrix2 * a}\n")

if option == 4:
  validation()
  print("\nElement wise Matrix Multiplication:\n")
  print(f"The final matrix is:\n {matrix1 * matrix2}\n")

if option == 5:
  print("\nMatrix Multiplication:\n")
  if matrix1.shape[1] != matrix2.shape[0]:
    print("Error - Matrix1 column must be equal to Matrix2 row count")
    quit()
  else:
    print(f"The final matrix is:\n {np.dot(matrix1,matrix2)}")

if option == 6:
  print("\nMatrix Transpose:\n")
  choice = input("Choose between a)matrix1 and b)matrix2: ")
  if choice == 'a':
    print(f"The final matrix is:\n {matrix1.T}\n")
  if choice == 'b':
    print(f"The final matrix is:\n {matrix2.T}\n")

if option == 7:
  print("\nTrace of a Matrix:\n")
  choice = input("Choose between a)matrix1 and b)matrix2: ")
  if choice == 'a':
    print(f"The final matrix is:\n {np.trace(matrix1)}\n")
  if choice == 'b':
    print(f"The final matrix is:\n {np.trace(matrix2)}\n")

if option == 8:
  print("\nSolve System of Linear Equations:\n")
  if matrix1.shape[0] != matrix1.shape[1]:
    print("Matrix A must be square for solving linear equations.")
    quit()
  else:
    solution = solve(matrix1, matrix1)
```

```
    print("Solution of linear equations:\n")
    print(solution)

if option == 9:
  print("\nDeterminant:\n")
  choice = input("Choose between a)matrix1 and b)matrix2: ")
  if choice == 'a':
    if matrix1.shape[0] != matrix1.shape[1]:
      print("Matrix must be square for determinant calculation.")
      quit()
    else:
      print(f"The final matrix is:\n {np.linalg.det(matrix1)}\n")
  if choice == 'b':
    if matrix2.shape[0] != matrix2.shape[1]:
      print("Matrix must be square for determinant calculation.")
      quit()
    else:
      print(f"The final matrix is:\n {np.linalg.det(matrix2)}\n")

if option == 10:
  print("\nInverse:\n")
  choice = input("Choose between a)matrix1 and b)matrix2: ")
  if choice == 'a':
    print(f"The final matrix is:\n {np.linalg.inv(matrix1)}\n")
  if choice == 'b':
    print(f"The final matrix is:\n {np.linalg.inv(matrix2)}\n")


if option == 11:
  print("\nSingular Value Decomposition:\n")
  choice = input("Choose between a)matrix1 and b)matrix2: ")
  if choice == 'a':
    print(f"The final matrix is:\n {svd(matrix1)}\n")
  if choice == 'b':
    print(f"The final matrix is:\n {svd(matrix2)}\n")


if option == 12:
  print("\nEigen Value:\n")
  choice = input("Choose between a)matrix1 and b)matrix2: ")
  if choice == 'a':
    print(f"The final matrix is:\n {eig(matrix1)}\n")
  if choice == 'b':
    print(f"The final matrix is:\n {eig(matrix2)}\n")


if option == 13:
  print("\nSearch an Element:\n")
  ele = int(input("Enter the element to search for: "))
  choice = input("Choose between a)matrix1 and b)matrix2: ")
  if choice == 'a':
    indices = np.where(matrix1 == ele)
```

```
    if indices[0].size > 0:
      print("Element found at:")
      print(list(zip(indices[0], indices[1])))
    else:
      print("Element not found")

  if choice == 'b':
    indices = np.where(matrix2 == ele)
    if indices[0].size > 0:
      print("Element found at:")
      print(list(zip(indices[0], indices[1])))
    else:
      print("Element not found")



if option == 14:
  print("\nDifference of Sum of Upper and Lower Triangular Matrix:\n")
  choice = input("Choose between a)matrix1 and b)matrix2: ")
  if choice == 'a':
    print(f"The final matrix is:\n")
    upper = np.sum(np.triu(matrix1))
    lower = np.sum(np.tril(matrix1))
    print(upper - lower)
  elif choice == 'b':
    print(f"The final matrix is:\n")
    upper = np.sum(np.triu(matrix2))
    lower = np.sum(np.tril(matrix2))
    print(upper - lower)
  else:
    print("Error")

if option == 15:
  print("Exitting...")
  condition = False
```