Ex.No: 2 Matrix Multiplication

DATE:

REGISTER NUMBER : 212222040036

AIM:

Write a python program for matrix multiplication and inspect for failures.

Algorithm:

Algorithm:

1. Start the program.
2. Create empty list formatrix1, matrix2 and result.
3. Get the rows and columns count from the user.
4. Get the values of two matrix.
5. Perform matrix multiplication and store the answer in result.
6. Stop the program.

Program:

```
r1, c1 = input("Enter row and column count in matrix 1: ").split()
r2, c2 = input("Enter row and column count in matrix 2: ").split()

matrix1 = []
matrix2 = []
result = []

# Check if input values are numeric
if r1.isnumeric() and c1.isnumeric() and r2.isnumeric() and c2.isnumeric():
    r1 = int(r1)
    r2 = int(r2)
    c1 = int(c1)
    c2 = int(c2)

    # Check if matrix multiplication is possible
    if c1 != r2:
        print("Matrix multiplication not possible")
    elif max(r1, c1, r2, c2) > 20 or min(r1, c1, r2, c2) == 0:
        print("Matrix multiplication not possible")
    else:
        # Input matrix 1

        print("Enter matrix 1:")
        for i in range(r1):
            a = []
            for j in range(c1):
                a.append(int(input(f"Element [{i+1}][{j+1}] for matrix 1: ")))
            matrix1.append(a)

        # Input matrix 2
        print("Enter matrix 2:")
        for i in range(r2):
            a = []
            for j in range(c2):
                a.append(int(input(f"Element [{i+1}][{j+1}] for matrix 2: ")))
            matrix2.append(a)
```

```
        # Matrix multiplication logic
        for i in range(r1):
            inter = []
            for j in range(c2):
                sum = 0
                for k in range(c1):  # Should be c1 because it matches r2
                    sum += matrix1[i][k] * matrix2[k][j]
                inter.append(sum)
            result.append(inter)

        # Output the result
        print("Resultant matrix after multiplication:")
        for i in range(r1):
            for j in range(c2):
                print(result[i][j], end=" ")
            print()

else:
    print("Enter valid numeric values for matrix dimensions.")
```

Output:

```
Enter the size of a: 2 3
Enter the size of b: 2 3
Matrix multiplication is not possible.
Reason to fail: to do multiplication of matrices the number of columns in matrix –a[] should be
equal to the number of rows in matrix in b[]


Enter the size of a: p q
Enter the size of b: q s
Matrix multiplication is not possible.
Reason to fail: to do multiplication of matrices the number of columns in matrix –a[]
should be equal to number of rows in matrix –bl, and rows & columns should be integer
values.



Enter the size of a: 1.5 2
Enter the size of b: 2 3
Matrix multiplication is not possible.
Reason to fail: to do multiplication of matrices the number of columns in matrix –al should be
equal to number of rows in matrix –bl, and rows & columns should be integer values.


Enter the size of a: 350
480 Enter the size of b:
480 620
Matrix multiplication is not possible.
Reason to fail: size of buffer will be not be sufficient to handle this multiplication.
```

```
Enter the size of a: -
1 -2 Enter the size of
b: -2 3
Matrix multiplication is not possible.
Reason to fail: to do multiplication of matrices the number of columns in matrix –al should
be equal to number of rows in matrix –bl, and rows & columns should be positive integer values.
```

Result:

Thus, the python program for matrix multiplication is implemented and the causes for its failure is introspected successfully.