Adamson University
College of Engineering
Computer Engineering Department

Linear Algebra

Laboratory Activity No. 7

Matrix Algebra

Submitted by:
Columba, Lorenzo Miguel L.

Instructor:
Engr. Dylan Josh D. Lopez

December 21, 2020

I. Objectives

This laboratory activity aims to implement the principles and techniques of applying the operations in solving intermediate equations and applying matrix algebra in engineering solutions. Also, to be familiar with the fundamentals of the matrix operations.

II. Methods

The practices of this activity are to understand matrix algebra with its fundamentals of matrix operations. This activity implies and teaches the techniques of the operation such as transposition, dot product, determinants, and inverse. The deliverables of this activity are to perform matrix operations and prove the properties of the dot product. It should prove that the equivalence of the property could be True or False.

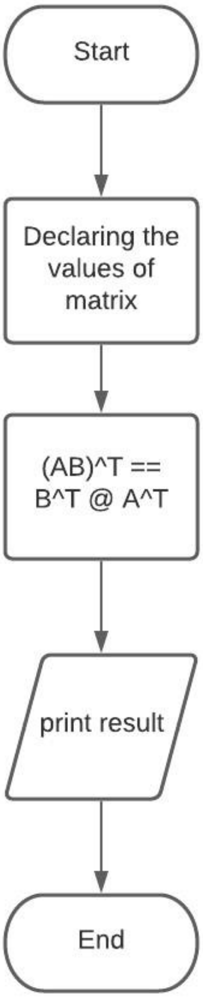

Figure 1: First property

Figure 1 shows the flow chart for the first property of the dot product which is the transposition property, this property states that a transpose matrix A and B is not equivalent to its dot product between a matrix of a transpose matrix B and A.

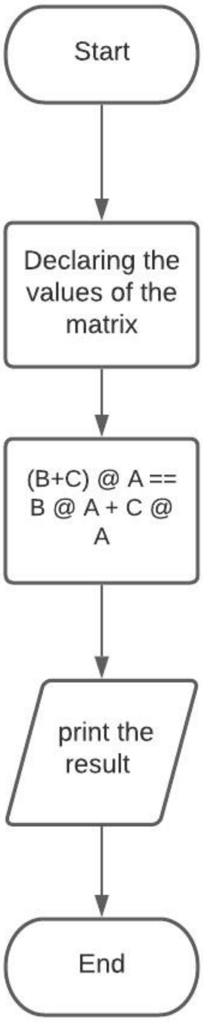

Figure 2: Second property

Figure 2 shows the flow chart of the second dot product property which is the distributive property, it explains that the dot product of matrix A and the result of the sum of matrix B and C is equivalent to the dot product of matrix B and A adding the result of the dot product of matrix C and A.

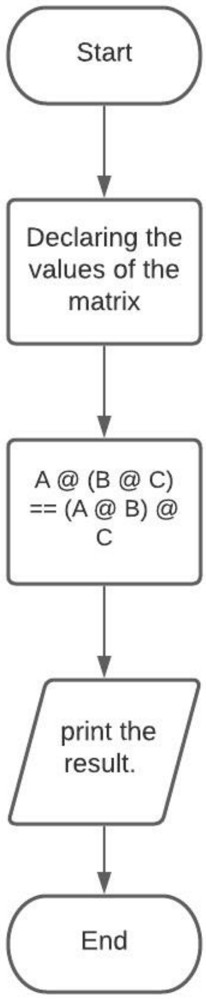

Figure 3: Third Property

Figure 3 shows the flow chart of the dot property which is the associative property that explains the dot product of matrix A and the result of the matrix B and C is equivalent to the dot product of matrix A and B then the matrix C will be multiplied to its dot product to the result of the matrix A and B.

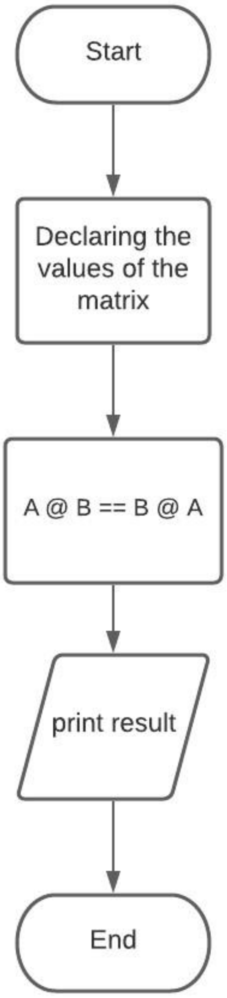

Figure 4: Fourth property

Figure 4 shows the flowchart of the dot product property which is not commutative, explains the dot property of matrix A and B is equivalent to the dot product of matrix B and A.

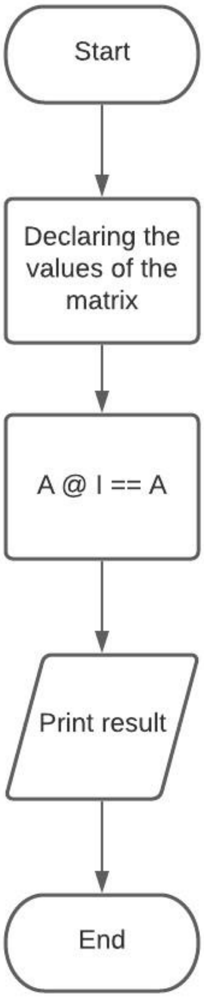

Figure 5: Fifth property

Figure 5 shows the flow chart of the fifth property which is identity, it expalains that the dot product of matrix A and the identity of I is equivalent to itself which is matrix A.

III. Results

In this activity the instructore required use to create a 6 matrix properties and prove and implement its properties which is not lower than a shape of (3,3).

```
mtrx_A = np.array([
    [1, 2, 1],
    [2, 0, 2],
    [2, 1, 1]
])
mtrx_B = np.array([
    [1,1,2],
    [3,3,1],
    [-1,-2,0]
])
mtrx_C = np.array([
    [0,1,1],
    [1,1,2],
    [3,3,1]
])
print(A.shape)
print(B.shape)
print(C.shape)
```

Figure 6: Code snippet of property 1.

In figure 6 shows the code snippet of a transposition property of the dot product which is done by flipping the values of its elements over its diagonals. The rows and columns from the original matrix will be switch.

```
B @ A
```

```
array([[ 7,  4,  5],
       [11,  7, 10],
       [-5, -2, -5]])
```

```
np.array_equiv(A @ B , B @ A)
```

```
False
```

Figure 7: Output of the figure 6

Figure7 shows the proof of results from figure 7 that the property is right which it shows the transpositions of the values of its elements which proves by displaying the equivalence of the matrices is False because its not equal. Using np.array_equiv() to get the result if it has the same result to each other.

2. $A \cdot (B \cdot C) = (A \cdot B) \cdot C$

```
[ ] A @ (B @ C)

array([[17, 23, 20],
       [10, 10,  0],
       [18, 22, 15]])
```

```
[ ] (A @ B) @ C

array([[17, 23, 20],
       [10, 10,  0],
       [18, 22, 15]])
```

Figure 8: Code Snippet of property 2

Figure 8 shows the code snippet of the second property which is distributive property, I declared three different matrices, to determine its value I code it one by one to know its values, After that I solve the whole equation to see if the property and its values is the same.

```
[ ] np.array_equiv((A @ (B @ C)),((A @ B)@C) )

True
```

Figure 9: Output of figure 8

Figure 9 shows the proof of the distributive property and with its values has the same equivalent to each other that proves it by printing its equivalence as True.

3. $A \cdot (B + C) = A \cdot B + A \cdot C$

```
[ ] A @ (B + C)

array([[11, 11, 10],
       [ 6,  6,  8],
       [ 8,  9, 10]])

[ ] A @ B + A @ C

array([[11, 11, 10],
       [ 6,  6,  8],
       [ 8,  9, 10]])
```

Figure 10: Code snippet for property 3

Figure 10 shows the code snippet of the associative property, to determine its values I solve each matrices in the equation and apply the dot product operation. After that I solve the whole equation number 3 to validate if it has the same result to each other by displaying a True or False.

```
[ ] np.array_equiv(A @ (B + C),A @ B + A @ C )

True
```

Figure 11: Output of figure 11

Figure 11 shows the result in proving that the equivalence of the matrices is the same and the property is True.

4. $(B+C)\cdot A = B\cdot A + C\cdot A$

```
(B+C) @ A
```

```
array([[11,  5,  8],
       [18, 11, 15],
       [ 6,  5,  5]])
```

```
[ ]  B @ A + C @ A
```

```
array([[11,  5,  8],
       [18, 11, 15],
       [ 6,  5,  5]])
```

Figure 12: Code snippet of property 4

Figure 12 shows the code snippet of Not Commutative property, to determine its values I solve matrix A and B separately to get its dot product result and to validate if the equivalence of matrices are the same.

```
[ ]  np.array_equiv((B+C) @ A , B @ A + C @ A)
```

```
True
```

Figure 13: Output of figure 13

Figure 13 shows the result of proof in the property 4 that displays as true meaning it this property is equal values.

```
[ ] I = np.linalg.inv(A)
    I

    array([[-0.5 , -0.25,  1.  ],
           [ 0.5 , -0.25,  0.  ],
           [ 0.5 ,  0.75, -1.  ]])

[ ] A @ I

    array([[1., 0., 0.],
           [0., 1., 0.],
           [0., 0., 1.]])

[ ] A

    array([[1, 2, 1],
           [2, 0, 2],
           [2, 1, 1]])
```

Figure 14: Code snippet of property 5

Figure 14 shows the identity property which states that when 1 is multiplied by any real number, the number does not change; that is, any number times 1 is equal to itself. The number "1" is called the multiplicative identity for real numbers. [1]

```
[ ] np.allclose(np.dot(A,I), np.eye(3))

    True
```

Figure 15: Output of figure 14

Figure 15 shows the proof of the reults of the identity property which it has the same equivalence to the martrix A that proves by printing a True array.

IV. Conclusion

This laboratory helps me to understand the implementations and techniques in performing the fundamental operation in matrix algebra such as transposition, dot product or inner product, determinant, and inverse. Also I familiarized the different usage of the NumPy's functions and infix operation in matrix multiplication. In addition, I achieved in this laboratory is manipulate and try to explore more in the different fundamentals operation in matrix algebra it helps me to understand how matrices works into different properties.

Lastly, matrix operation is also used in solving problems in a health care field they used matrix operation to determined the system equations in medical tomorgprahic imaging which images are artificially discretized with its approximations to the parts of the interior of a body. [2]. Also another book I've read is that matrix operations are used to silve the monthly health care payments in a hospital [3].

References

[1] Python Software Foundation, (2020). *About "A dedicated infix operator for matrix multiplication."* Available: https://docs.python.org/3/whatsnew/3.5.html#whatsnew-pep-465 Date Accessed : 12/15/2020

[2] ijitr.com(2020), *About "ROLE OF LINEAR ALGEBRA AND ITS APPLICATIONS IN HEALTH CARE MONITORING"* Available: http://www.ijitr.com/index.php/ojs/article/view/286 Date Accessed :

[3] Classzone, (2020). *About "Matrix Operations."* Available: https://www.classzone.com/eservices/home/pdf/student/LA204AAD.pdf Date Accessed : 12/15/2020