Adamson University
College of Engineering
Computer Engineering Department

Linear Algebra

Laboratory Activity No. 7

Matrix Operations

Submitted by:
Guy, Lawrence Adrian B.

Instructor:
Engr. Dylan Josh D. Lopez

December 21, 2020

I. Objectives

The laboratory activity aims to be familiar with the fundamental matrix operations. It also aims to introduce special properties of dot product. Using the knowledge gained, the students would be able to solve equations and create engineering solutions using matrices.

II. Methods

The practices of the activity are to perform matrix operations used in the last activity and more operations. The other operations that are not included in the last activity are transposition, determinant, and inverse.

The deliverable of the activity is to perform the given operations in the activity and prove the given 6 matrix multiplication properties. These are achieved by coding the matrix operations in Python.

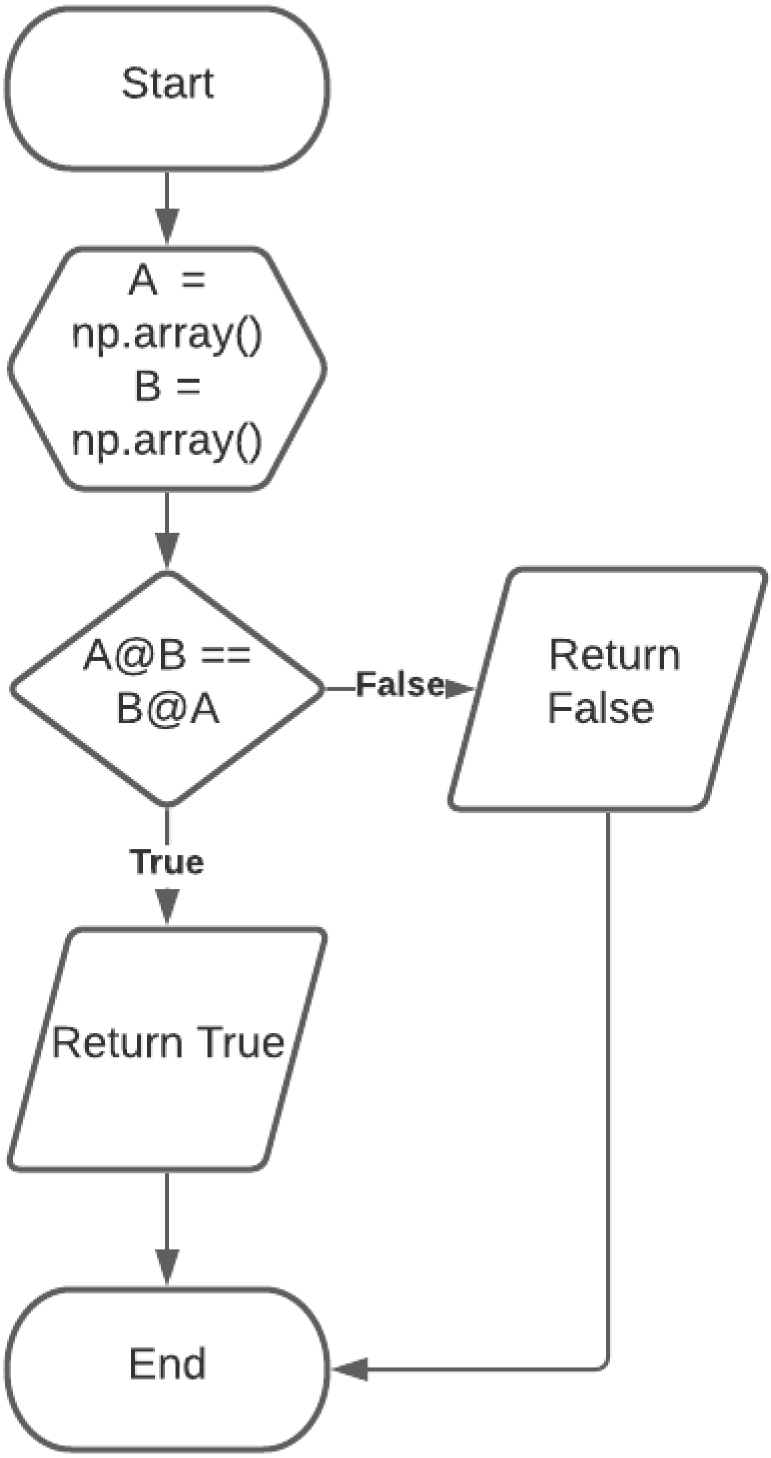

Figure 1 Flowchart for A@B!= B@A

The first property states that matrix multiplication or dot product of a matrix isn't commutative. Unlike in the addition of matrices, the result is not the same when the position of the matrices being multiplied is interchanged.

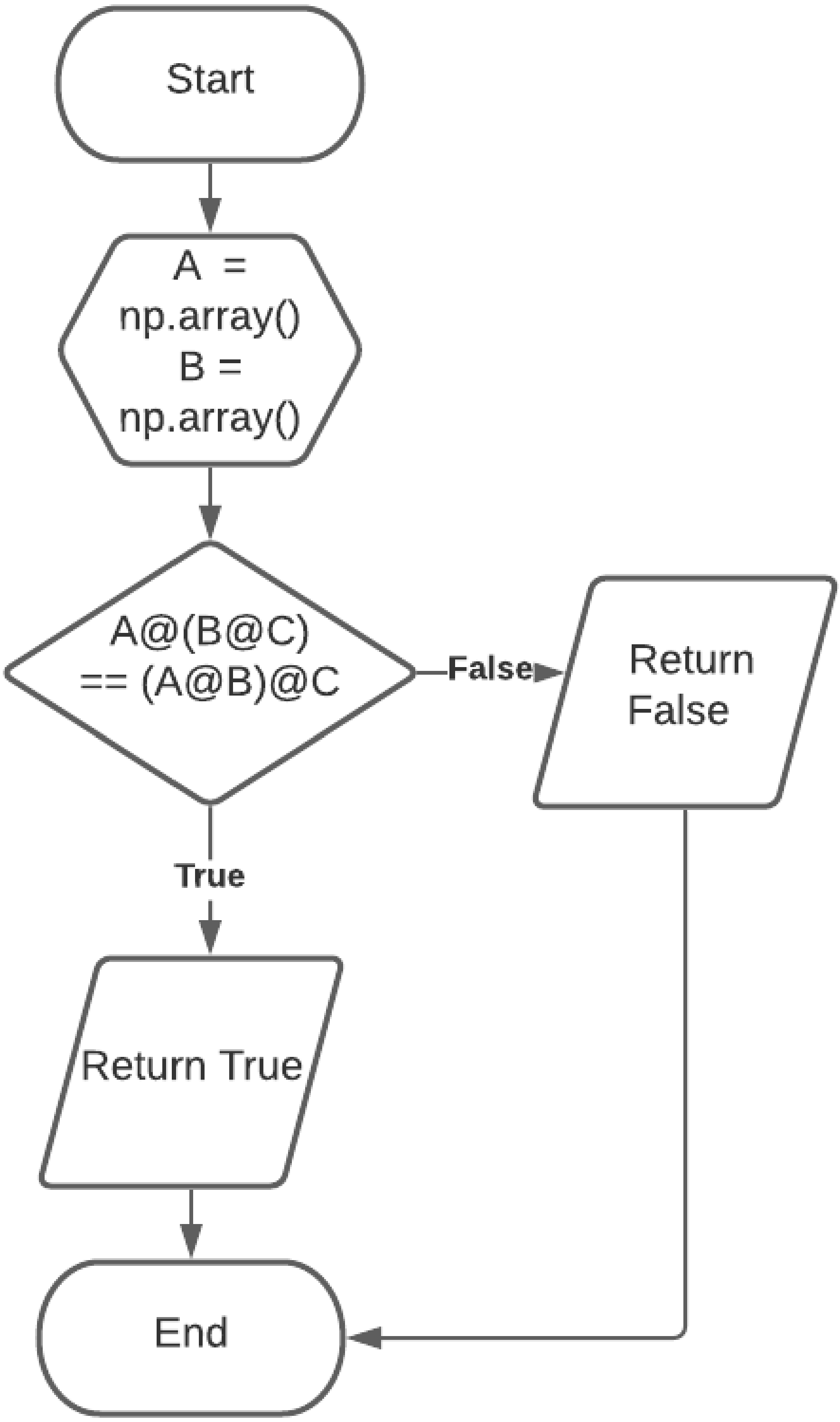

Figure 2 Flowchart for A@(B@C) == (A@B)@C

The second property is called the Associative Property of Multiplication. The property shows that the grouping of the multiplication of the matrix doesn't matter. The result will be the same whether A@B or B@C is done first. The result will stay the same.

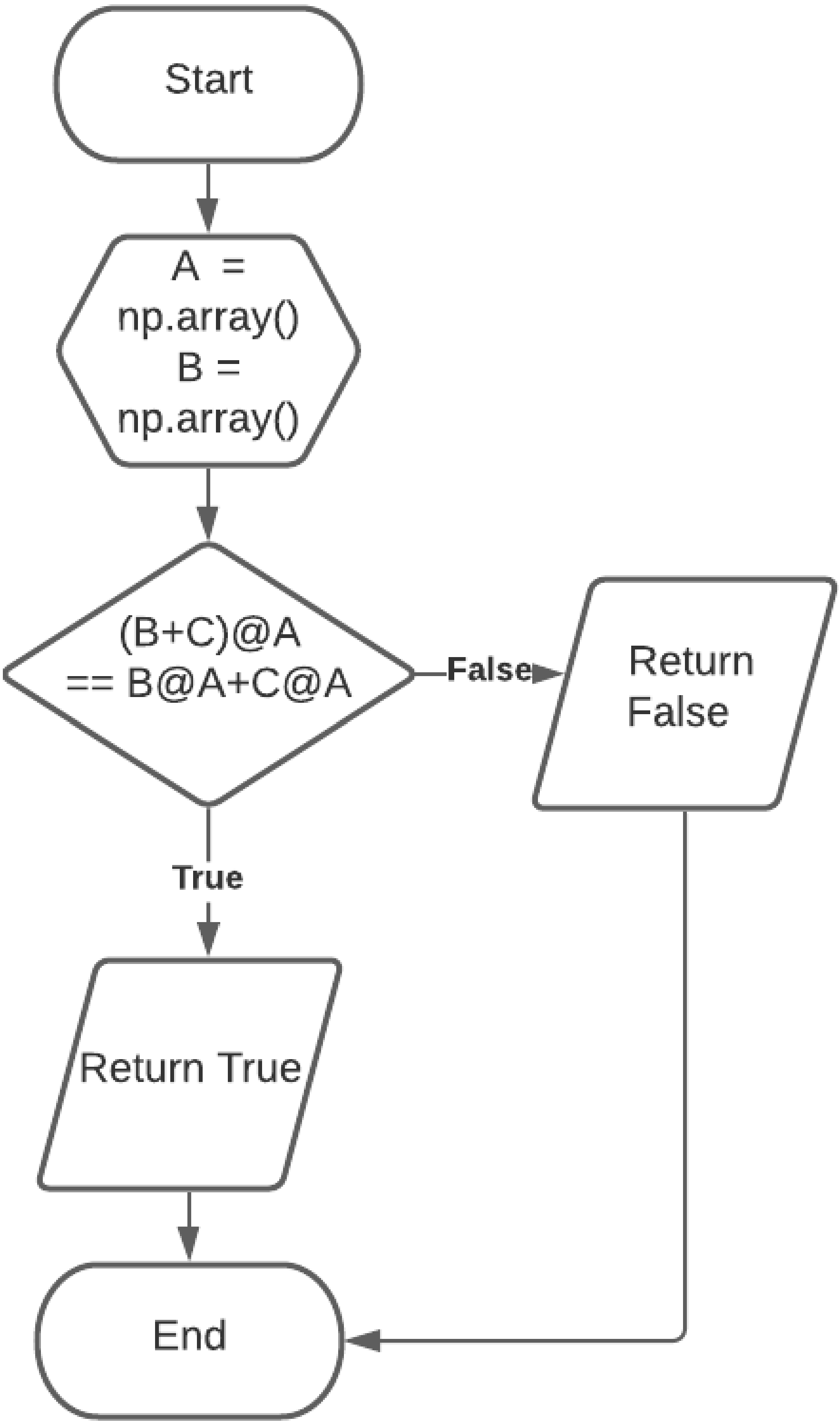

Figure 3 Flowchart for (B+C)@A == B@A+C@A

The fourth property is the Distributive Property. This property proves that matrices can be distributed before performing addition or subtraction and the result will be the same.

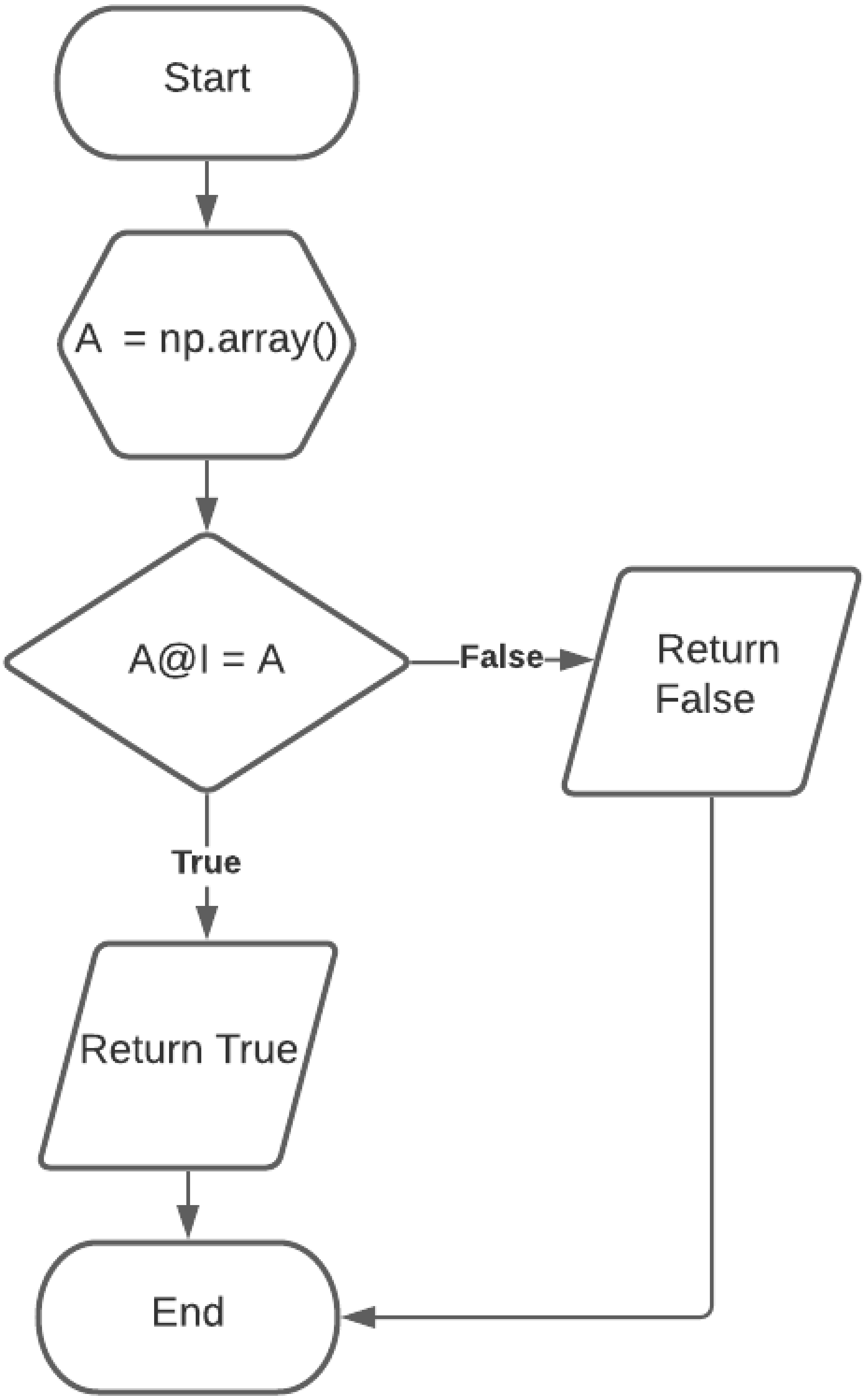

Figure 4 Flowchart for A@I = A

The fifth property is the Matrix Identity Property. This property states that when matrix A is multiplied to an identity matrix the answer is the matrix A.

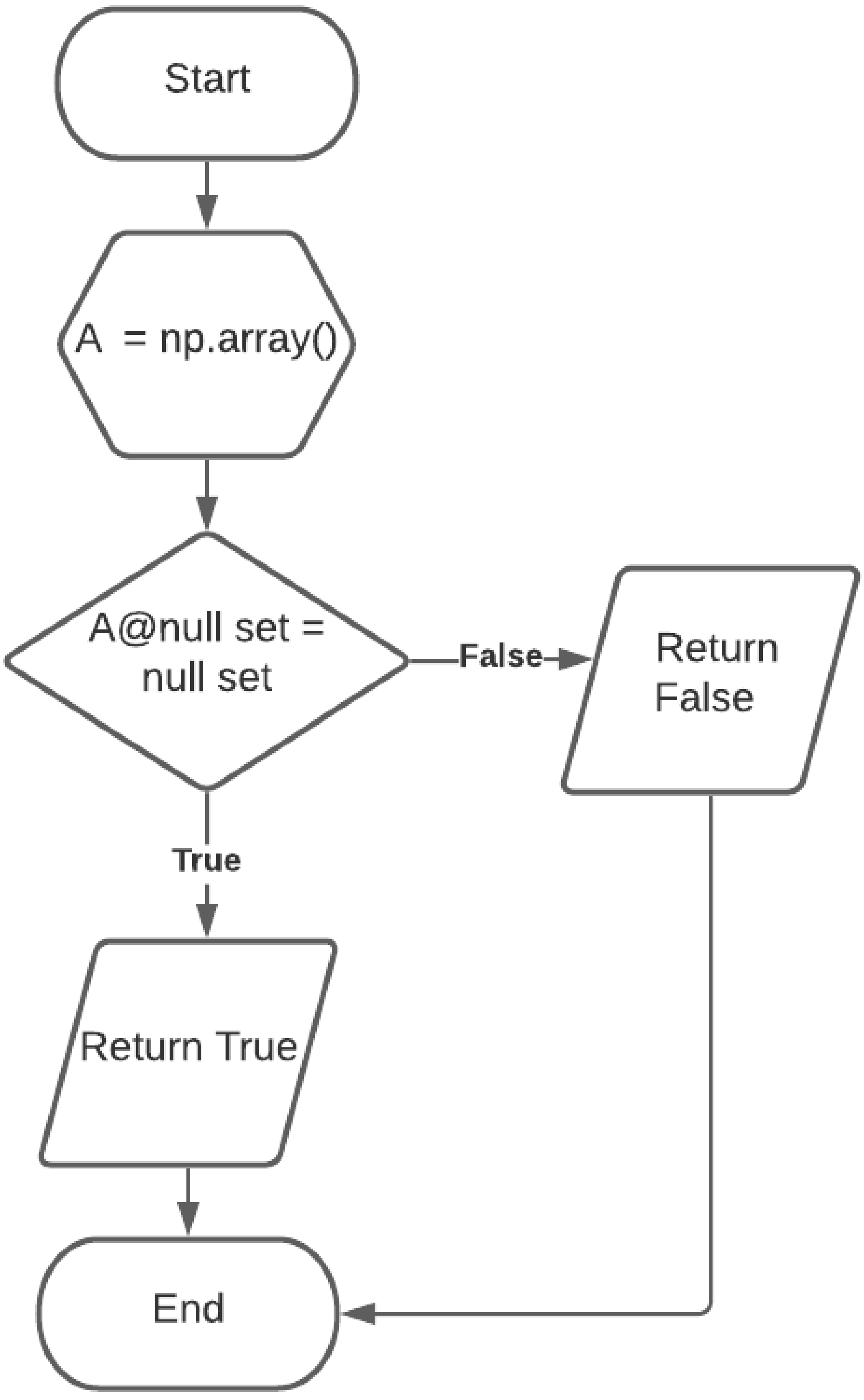

Figure 5 Flowchart for A@null = null

The sixth property is the zero or null property. This property states that when matrix A is multiplied to a null or zero matrix, the answer would be zero a zero or null matrix.

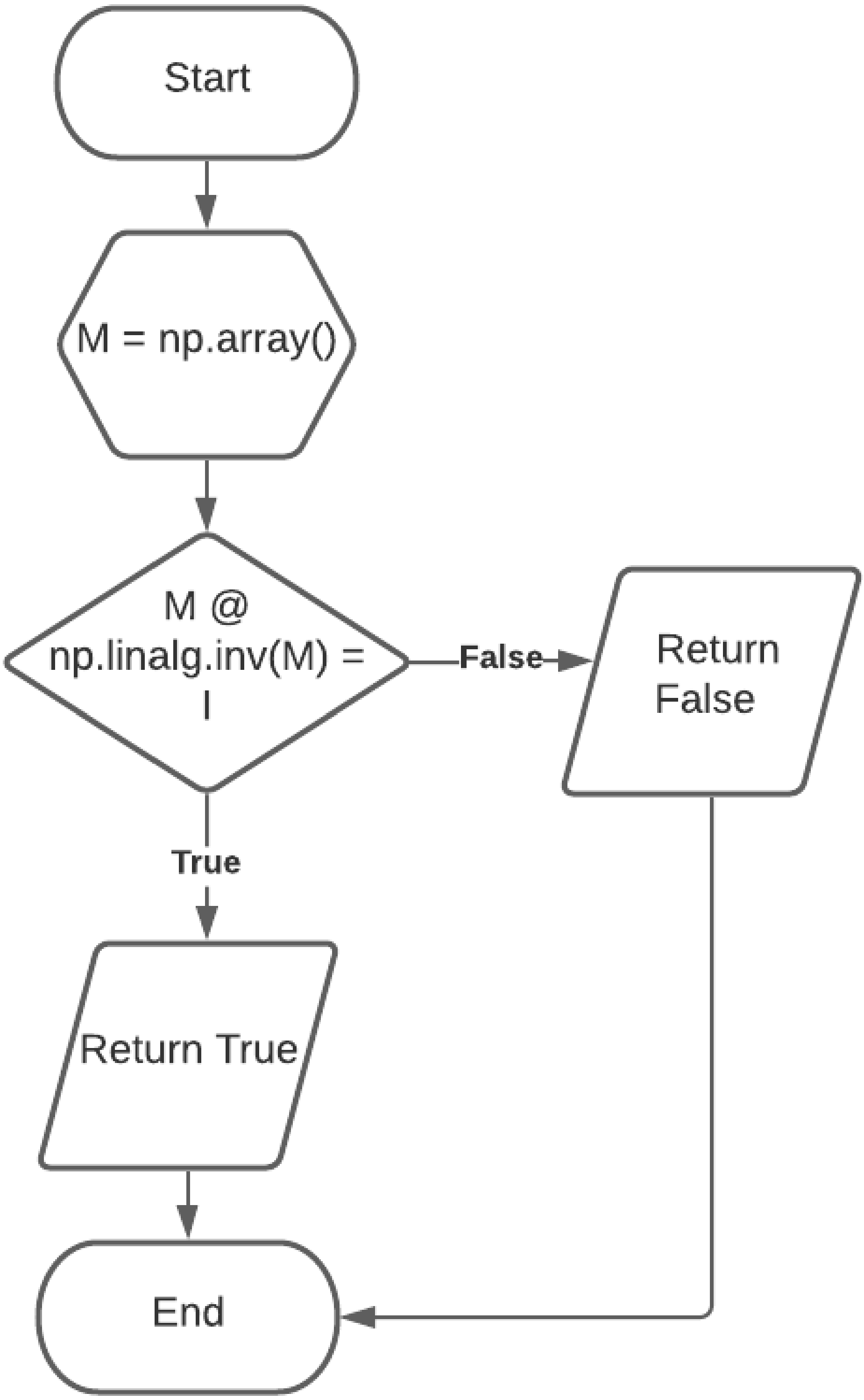

Figure 6 $M@M^{-1} = I$

The seventh property is the inverse property. This property states that when a square matrix is multiplied to its inverse the result would be an identiy matrix [1].

III. Results

1. $A \cdot B \neq B \cdot A$

```
A = np.array([
              [1,5,6],
              [0,0,1],
              [3,4,5]
])
B = np.array([
              [1,2,8],
              [1,0,1],
              [2,5,7]
])
```

```
print(A@B, "\n \n" , B@A)
print(A@B == B@A)
```

```
[[18 32 55]
 [ 2  5  7]
 [17 31 63]]

 [[25 37 48]
 [ 4  9 11]
 [23 38 52]]
[[False False False]
 [False False False]
 [False False False]]
```

Figure 7 Codes and output for the first property.

Figure 7 shows the codes and output of the first property. As seen in the output, the result of A@B is different compared to B@A. This proves that matrices aren't commutative.

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

```
A = np.array([
              [1,5,6],
              [0,0,1],
              [3,4,5]
])
B = np.array([
              [1,2,8],
              [1,0,1],
              [2,5,7]
])
C = np.array([
              [0,0,7],
              [4,1,2],
              [3,2,3]
])
```

```
print((A@(B@C)),"\n","\n",(A@B)@C)
print(A@(B@C) == (A@B)@C)
```

```
[[293 142 355]
 [ 41  19  45]
 [313 157 370]]

 [[293 142 355]
 [ 41  19  45]
 [313 157 370]]
[[ True  True  True]
 [ True  True  True]
 [ True  True  True]]
```

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

Figure 8 Codes and output for the second property

Figure 8 shows the codes and output of the second property. As seen in the output the result is the same in the dot product A@(B@C) and (A@B)@C. This proves that matrices are associative. The order of the multiplication shouldn't be changed because matrices aren’t commutative it would change the result.

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

```
A = np.array([
              [1,5,6],
              [0,0,1],
              [3,4,5]
])
B = np.array([
              [1,2,8],
              [1,0,1],
              [2,5,7]
])
C = np.array([
              [0,0,7],
              [4,1,2],
              [3,2,3]
])
```

```
print((B+C)@A,"\n\n", B@A+C@A)
print((B+C)@A == B@A+C@A)
```

```
[[46 65 83]
 [14 37 46]
 [35 65 87]]

 [[46 65 83]
 [14 37 46]
 [35 65 87]]
[[ True  True  True]
 [ True  True  True]
 [ True  True  True]]
```

Figure 9 Codes and output for the fourth property

Figure 9 shows the codes and output for the fourth property. As seen in the output, the result in the left-hand value when matrix A isn't distributed is the same as the right-hand value when matrix A is distributed to matrix B and C. This proves that matrices are distributive.

5. $A \cdot I = A$

```
A = np.array([
              [1,5,6],
              [0,0,1],
              [3,4,5]
])
```

```
print(A)
print(np.eye(3))
print(A@np.eye(3))
print(A@np.eye(3) == A)
```

```
[[1 5 6]
 [0 0 1]
 [3 4 5]]
[[1. 0. 0.]
 [0. 1. 0.]
 [0. 0. 1.]]
[[1. 5. 6.]
 [0. 0. 1.]
 [3. 4. 5.]]
[[ True  True  True]
 [ True  True  True]
 [ True  True  True]]
```

Figure 10 Codes and output for the fifth property

Figure 10 shows the codes and output for the fifth property. The dot product between a matrix and an identity matrix is the matrix itself. As seen in the output, when matrix A is multiplied to an identity matrix, the result is the same as matrix A. This is the only time when matrices can be multiplied in any order. May it be A@I or I@A.

6. $A \cdot \varnothing = \varnothing$

```
A = np.array([
                [1,5,6],
                [0,0,1],
                [3,4,5]
])

null_mat = np.zeros(3)
```

```
print(A)
print(null_mat)
print("Dot product:",A@null_mat)
print(np.array_equal(A@null_mat,null_mat))
```

```
[[1 5 6]
 [0 0 1]
 [3 4 5]]
[0. 0. 0.]
Dot product: [0. 0. 0.]
True
```

Figure 11 Codes and output for the sixth property

Figure 11 shows the codes and output for the sixth property. The sixth property shows that when a matrix is multiplied to a null/zero matrix, the answer is a null/zero matrix. As seen in the output in figure 11, it shows that the output is a zero matrix. This is further proven by using np.array_equal which checks if the dot product of a null/zero matrix and a matrix is equal to a null/zero matrix and the result is True.

7. $M \cdot M^{-1} = I$

```
M = np.array([
              [1,2,4],
              [3,3,0],
              [1,1,2]

])
```

```
print(M)
print(np.linalg.inv(M))
print(M@np.linalg.inv(M))
print(np.eye(3))
print(M@np.linalg.inv(M) == np.eye(3))
```

```
[[1 2 4]
 [3 3 0]
 [1 1 2]]
[[-1.          0.          2.        ]
 [ 1.          0.33333333 -2.        ]
 [ 0.         -0.16666667  0.5       ]]
[[1. 0. 0.]
 [0. 1. 0.]
 [0. 0. 1.]]
[[1. 0. 0.]
 [0. 1. 0.]
 [0. 0. 1.]]
[[ True  True  True]
 [ True  True  True]
 [ True  True  True]]
```

8. $(m, n) \cdot (n, k) = (m, k)$

Figure 12 Codes and output for the seventh property

Figure 12 shows the codes and output for the seventh property. The seventh property shows that the dot product of a matrix and its inverse is an identity matrix. As seen in the output, the matrix M is multiplied to its inverse. The inverse of the matrix is taken by using the function np.linalg.inv(). The result proves that the dot product of matrix M and its inverse is an identity matrix.

IV. Conclusion

The laboratory activity discussed matrix operations and properties of the dot product of matrices. The activity showed different ways of proving dot product properties. Through the activity, I was able to learn that the matrix properties and number properties are similar except for the commutative property. In number properties, numbers are commutative in multiplication but in matrix multiplication, matrices aren't commutative.

Matrix operations can be used in the healthcare field. One of its uses is to calculate the dosage needed for a patient. Doctors and nurses use linear equations when calculating the medical doses that are given to patients. Using linear equations, overdosage can be prevented especially for those patients that are taking multiple medicines at a time [2].

References

[1] mathisfun, “Inverse of a Matrix,” 2017. https://www.mathsisfun.com/algebra/matrix-inverse.html (accessed Dec. 19, 2020).

[2] C. D. Crowder, “What Careers Use Linear Equations?,” 2018. https://sciencing.com/careers-use-linear-equations-6060294.html (accessed Dec. 19, 2020).

Appendix

Github Repository Link:

https://github.com/Loreynszxc/Linear-Algebra-Lab-7