Adamson University
College of Engineering
Computer Engineering Department

Linear Algebra

Laboratory Activity No. 4

Vector Operations

Submitted by:

Roa, Danica Kate I.

Instructor:

Engr. Dylan Josh D. Lopez

November, 04, 2020

I. Objectives

This laboratory activity aims to implement the principles and techniques of vector operation. This also aims to be familiarize on new operation such as products and to visualize the vector operation.

II. Methods

The acts of this laboratory is to acclimate on utilizing the vector activity and to envision the vector activity utilizing the python language. Having this movement will show us on step by step instructions to actualize the vector activity. There are capacities that I have been utilized in a lab action. The import math in python offers admittance to do the numerical cycle with the end goal that square root, geometrical capacities, and so forth Math.sqt is utilized to restore the buoy esteem. The def speck is utilized to increase the two given qualities. The import matplotlib.pyplot as plt is an assortment of order style works that matplotlib takes a shot at. Each pyplot cycle changes part of the picture: for instance, make a shape, make a plot region on the figure, attract a few lines the plot region, enrich the way with names, and so forth Python abs() is an underlying capacity with standard Python library accessible. It gives the outright incentive for a given number. The aggregate sum of a number is its worth paying little mind to its sign. Numbers can be whole numbers, drifting point numbers or complex numbers. The fig.gca restores the estimation of the current axis where the "3d" is utilized to extend the axis.

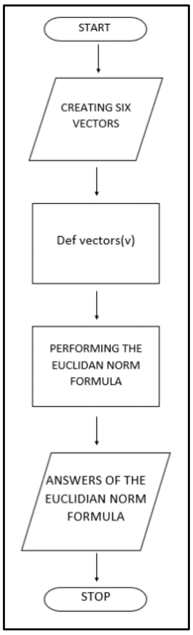

START
CREATING SIX VECTORS
Def vectors(v)
PERFORMING THE EUCLIDAN NORM FORMULA
ANSWERS OF THE EUCLIDIAN NORM FORMULA
STOP

Figure 1: Flowchart of Task 1

In figure one, it shows the cycle that I've been done on making the codes of assignment 1. From the outset, I made six exceptional vectors that will be utilized to settle the Euclidian Norm Equation. The def vectors are the one that is utilized to restore the estimation of the subsequent to playing out the Euclidian Norm Formula. It will print the estimations of the Euclidian Norm.

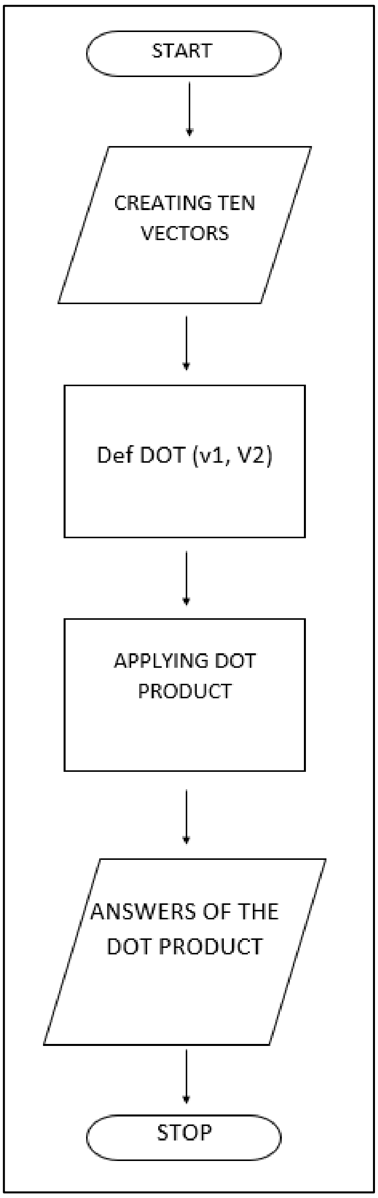


Figure 2: Flowchart of Task 2

Figure 2 is the flowchart for Task 2, where they speak to the cycle of how the codes went. From the start, I make ten exceptional vectors that will be utilized to the spot item. The def dot is the arithmetical activity where it restores the two vectors into a solitary. number. At that point it will print the profit esteems for the dab item subsequent to performing it.

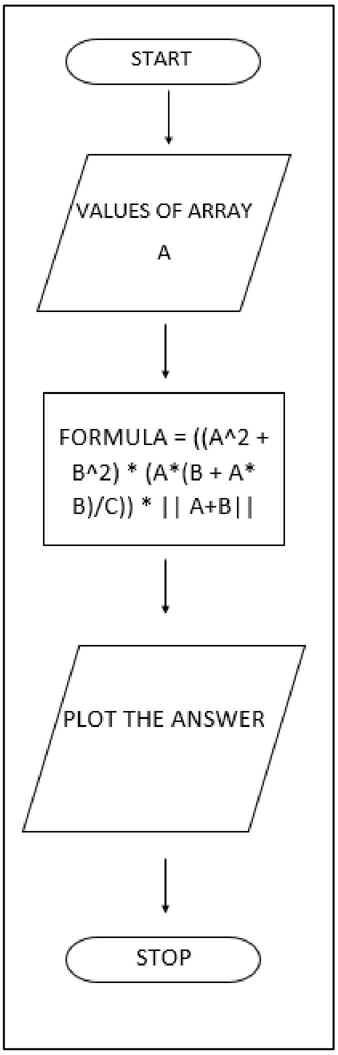
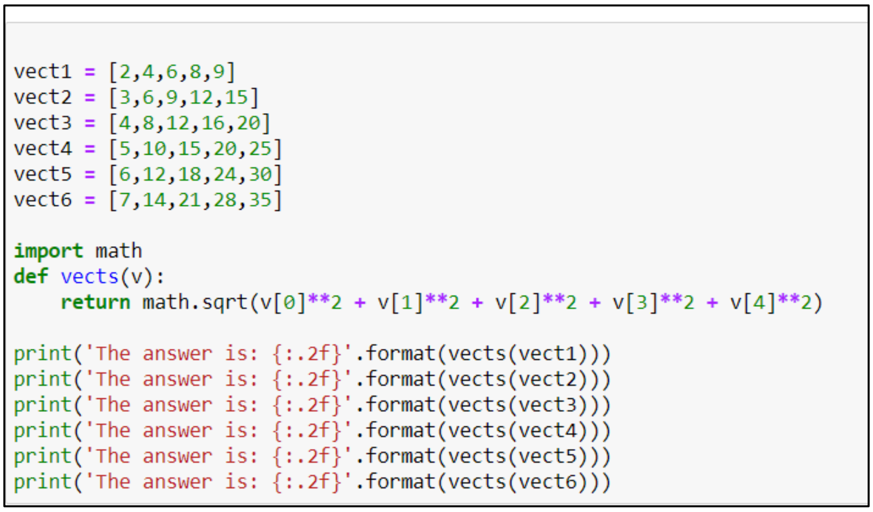


Figure 3: Flowchart of Task 3

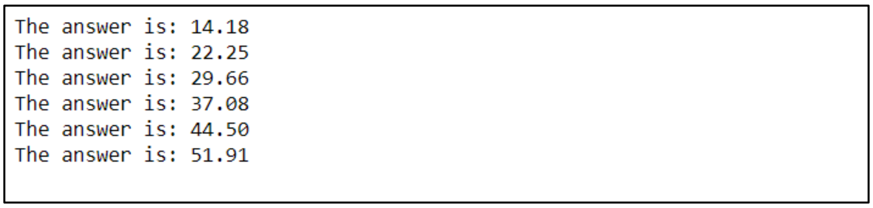
Figure 3 is the flowchart for task 3 is the portrayal of the way toward making the codes. I am contributing the estimations of A, which is given on the errand. Subsequent to putting the contribution of the exhibits that are given, I play out the formula $((\mathrm{A}^{2} + B^{2}) \cdot (\mathrm{A} * (\mathrm{B} + \mathrm{A} * \mathrm{B})./\mathrm{C})) * ||\mathrm{A} + \mathrm{B}||$ Having the appropriate response of the figured cluster, I plotted the appropriate response into three-measurement to without any problem envision the cluster.

III. Results



```
vect1 = [2,4,6,8,9]
vect2 = [3,6,9,12,15]
vect3 = [4,8,12,16,20]
vect4 = [5,10,15,20,25]
vect5 = [6,12,18,24,30]
vect6 = [7,14,21,28,35]

import math
def vects(v):
    return math.sqrt(v[0]**2 + v[1]**2 + v[2]**2 + v[3]**2 + v[4]**2)

print('The answer is: {:.2f}'.format(vects(vect1)))
print('The answer is: {:.2f}'.format(vects(vect2)))
print('The answer is: {:.2f}'.format(vects(vect3)))
print('The answer is: {:.2f}'.format(vects(vect4)))
print('The answer is: {:.2f}'.format(vects(vect5)))
print('The answer is: {:.2f}'.format(vects(vect6)))
```

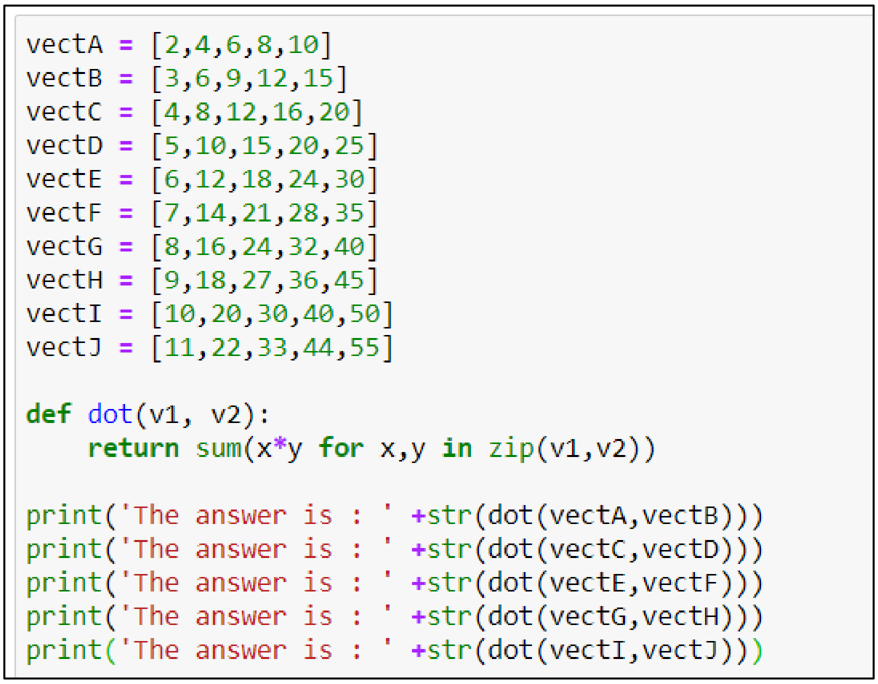
Figure 4 task one code



```
The answer is: 14.18
The answer is: 22.25
The answer is: 29.66
The answer is: 37.08
The answer is: 44.50
The answer is: 51.91
```

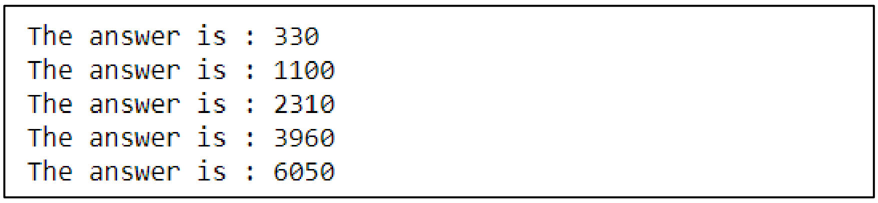
Figure 5 output of *Figure 4*

On the figure 4, I made six distinct vectors that has a five components. The import math is utilized to have an acess to do the numerical activity. The def vectors will bring that back estimation of the processed vectors where the Euclidian Norm has been performed. The yield is the contrast between the vectors(Figure 5).



```
vectA = [2,4,6,8,10]
vectB = [3,6,9,12,15]
vectC = [4,8,12,16,20]
vectD = [5,10,15,20,25]
vectE = [6,12,18,24,30]
vectF = [7,14,21,28,35]
vectG = [8,16,24,32,40]
vectH = [9,18,27,36,45]
vectI = [10,20,30,40,50]
vectJ = [11,22,33,44,55]

def dot(v1, v2):
    return sum(x*y for x,y in zip(v1,v2))

print('The answer is : ' +str(dot(vectA,vectB)))
print('The answer is : ' +str(dot(vectC,vectD)))
print('The answer is : ' +str(dot(vectE,vectF)))
print('The answer is : ' +str(dot(vectG,vectH)))
print('The answer is : ' +str(dot(vectI,vectJ)))
```

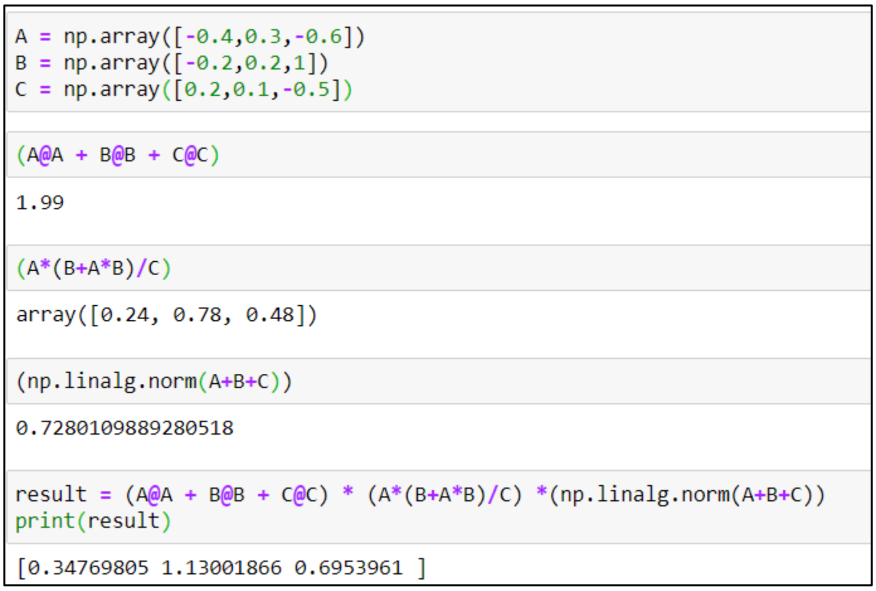
Figure 6 task 2 code



```
The answer is : 330
The answer is : 1100
The answer is : 2310
The answer is : 3960
The answer is : 6050
```

Figure 7 Output of *Figure 6*

For the Figure 6 , I have to make a 5 particular sets of vectors which is ten vectors that the components doesn't have lower than five. The def dot where the speck is the name of the fuction furthermore, the contention are the v1 and v2 with the goal that it return the estimations of the two vectors. The figure 7 is the result of the pair vectors.



```
A = np.array([-0.4,0.3,-0.6])
B = np.array([-0.2,0.2,1])
C = np.array([0.2,0.1,-0.5])
```

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

```
1.99
```

```
(A*(B+A*B)/C)
```

```
array([0.24, 0.78, 0.48])
```

```
(np.linalg.norm(A+B+C))
```

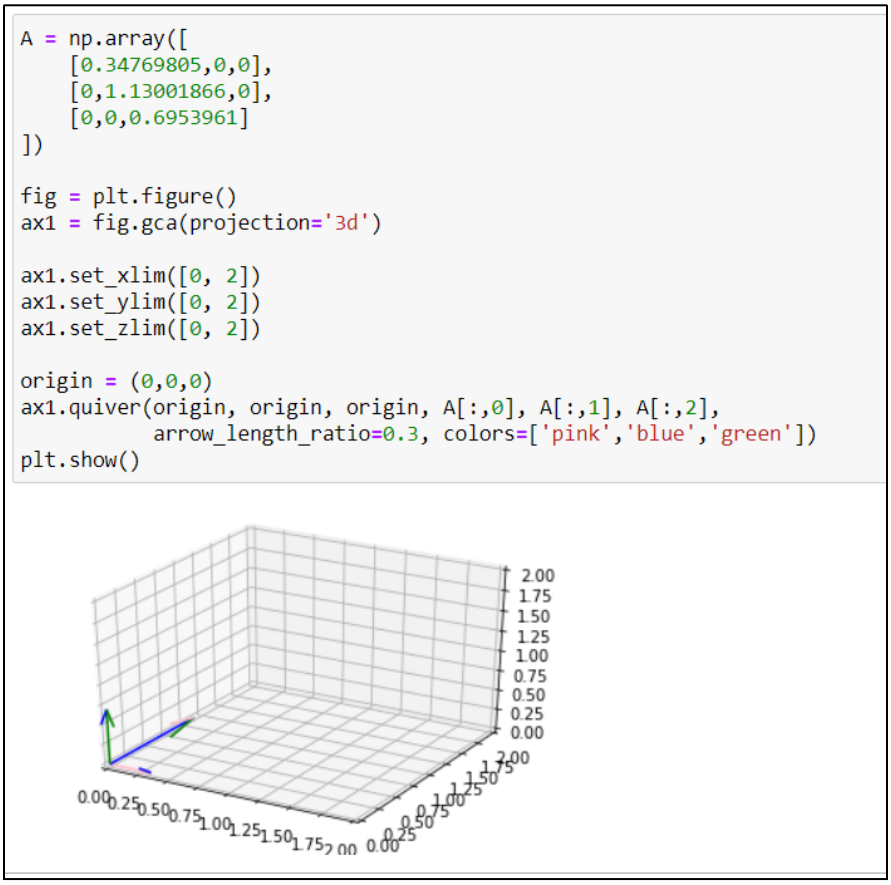
```
0.7280109889280518
```

```
result = (A@A + B@B + C@C) * (A*(B+A*B)/C) *(np.linalg.norm(A+B+C))
print(result)
```

```
[0.34769805 1.13001866 0.6953961 ]
```

Figure 8.1 Task 3

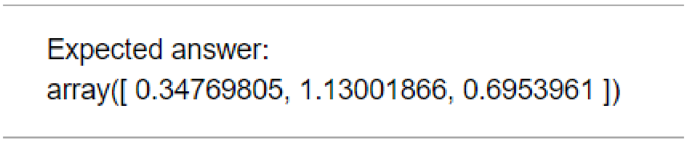
On the figure 8.1, it has a given vector inputs where the three array is made with the estimations of given vectors. The abs(A+B+C) is the total of the three vectors where the entirety will be a flat out worth. In running out the given equation the inputed value should be a absolute value. The expected and the value computed by the code.



```
A = np.array([
    [0.34769805,0,0],
    [0,1.13001866,0],
    [0,0,0.6953961]
])

fig = plt.figure()
ax1 = fig.gca(projection='3d')

ax1.set_xlim([0, 2])
ax1.set_ylim([0, 2])
ax1.set_zlim([0, 2])

origin = (0,0,0)
ax1.quiver(origin, origin, origin, A[:,0], A[:,1], A[:,2],
           arrow_length_ratio=0.3, colors=['pink','blue','green'])
plt.show()
```

Figure 8.2 Task 3 Continuation

As making a portrayal of the vectors, I make a cluster with the utilization of registered values. The plt.figure is the one that will make the figure of the three-measurement which will help us to see the array.



Expected answer:
array([0.34769805, 1.13001866, 0.6953961])

Figure 9 Task 3 Expected value

IV. Conclusion

Having these Laboratory will help us with understanding the vector activity. On this action we can investigate the various methods of controlling the vector activity. Like on the task 1 were we can utilize the euclidian standard and the np.linalg.norm that has a similar capacity which is to quantify the separation between the two vectors. These research center movement likewise show us how to imagine the vector activity and furthermore how to speak to these vectors into three-dimensional chart. The idea of these vector activity is to speak to the vector utilizing spot item and cross item in straight variable based math and these likewise speak to a certain dimensional space.

References

[1] D.J.D. Lopez. "Adamson University Computer Engineering Department Honor Code," AdU-CpE Departmental Policies, 2020.