

**LAPORAN PRAKTIKUM UTS
KECERDASAN BUATAN
“NEURAL NETWORKS”**



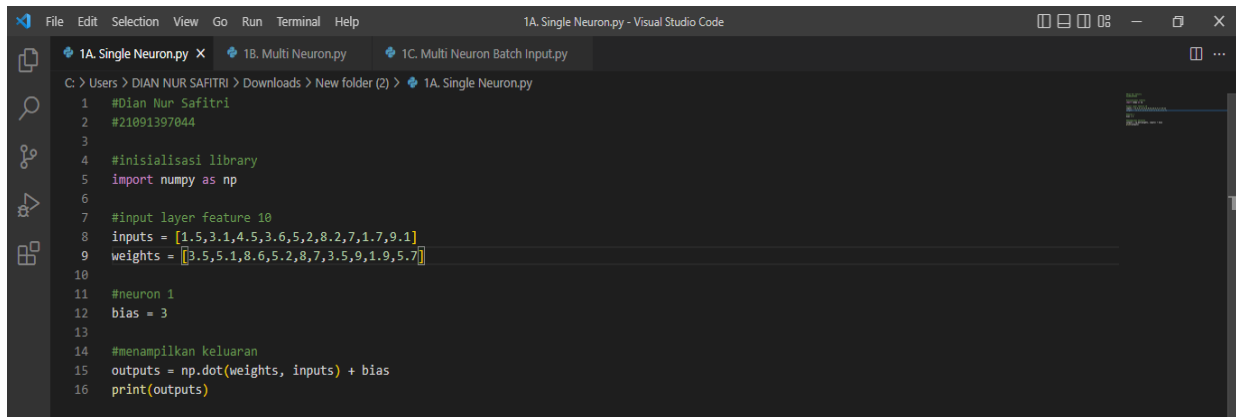
**Disusun oleh :
Dian Nur Safitri**

**NIM/Kelas :
21091397044/2021B**

**D4 MANAJEMEN INFORMATIKA
FAKULTAS VOKASI
UNIVERSITAS NEGERI SURABAYA
2022**

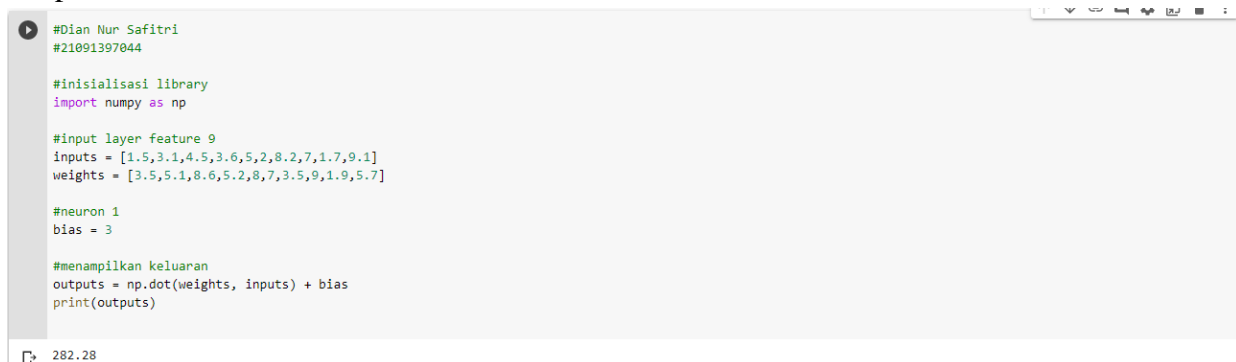
A. SOURCE CODE

1. Single Neuron



```
1A. Single Neuron.py - Visual Studio Code
1A. Single Neuron.py X 1B. Multi Neuron.py 1C. Multi Neuron Batch Input.py
C:\Users\DIAN NUR SAFITRI\Downloads\New folder (2)\> 1A. Single Neuron.py
1 #Dian Nur Safitri
2 #21091397044
3
4 #inisialisasi library
5 import numpy as np
6
7 #input layer feature 10
8 inputs = [1.5,3.1,4.5,3.6,5.2,8.2,7.1,7,9.1]
9 weights = [3.5,5.1,8.6,5.2,8,7,3.5,9,1.9,5.7]
10
11 #neuron 1
12 bias = 3
13
14 #menampilkan keluaran
15 outputs = np.dot(weights, inputs) + bias
16 print(outputs)
```

Output :



```
#Dian Nur Safitri
#21091397044

#inisialisasi library
import numpy as np

#input layer feature 9
inputs = [1.5,3.1,4.5,3.6,5.2,8.2,7.1,7,9.1]
weights = [3.5,5.1,8.6,5.2,8,7,3.5,9,1.9,5.7]

#neuron 1
bias = 3

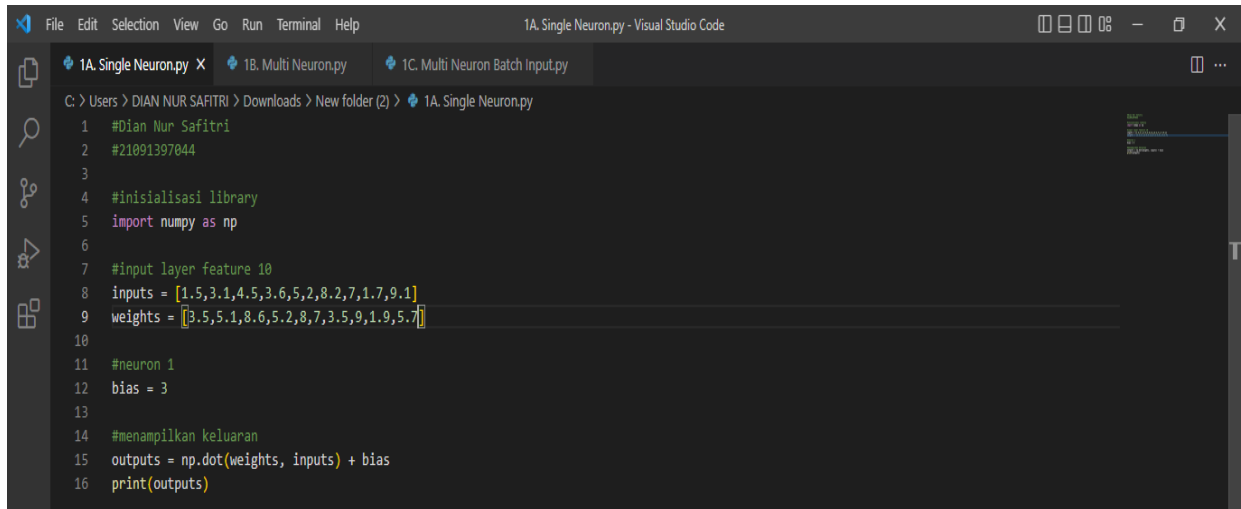
#menampilkan keluaran
outputs = np.dot(weights, inputs) + bias
print(outputs)

282.28
```

Penjelasan Pengerjaan :

- Import numpy merupakan library python
- Setiap neuron mempunyai sebuah koneksi yang unik ke setiap neuron, yang outputnya menjadi neuron yang inputnya kita kodekan.
- Setiap input unik memiliki bobot unik yang juga saling terkait. dikarenakan inputnya sebanyak 10, jadi perlu menyatakan bobot sebanyak 10
- Penambahan vektor akan digunakan untuk penambahan bias
- Setiap sisi neuron (inputs dan weights) unik memiliki bias yang unik juga. Jadi diinputkan bias sama dengan 3 atau sebagainya
- Langkah pertama :
Sebuah Neuron menjumlahkan semua inputs kali bobot ditambah bias, jadi sederhananya kita masukkan $outputs = np.dot(weights, inputs) + bias$
- Mencetak output, jalankan. `print (outputs)`
- Hasil output : 282.28

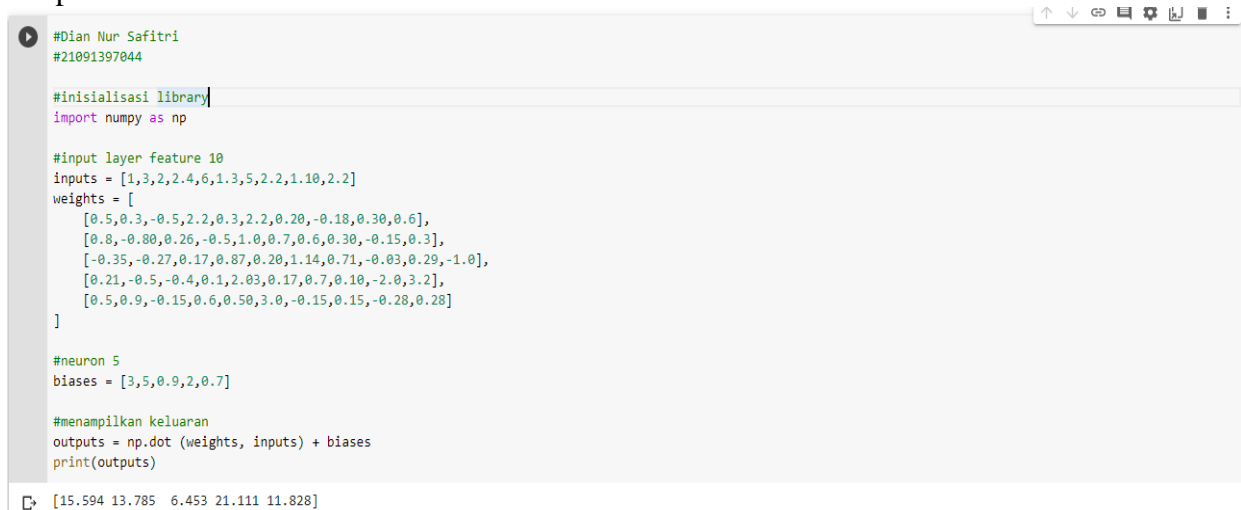
2. Multi Neuron



The screenshot shows a Visual Studio Code window titled "1A. Single Neuron.py - Visual Studio Code". The editor has three tabs: "1A. Single Neuron.py", "1B. Multi Neuron.py", and "1C. Multi Neuron Batch Input.py". The active tab "1A. Single Neuron.py" contains the following Python code:

```
C:\> Users > DIAN NUR SAFITRI > Downloads > New folder (2) > 1A. Single Neuron.py
1  #Dian Nur Safitri
2  #21091397044
3
4  #inisialisasi library
5  import numpy as np
6
7  #input layer feature 10
8  inputs = [1.5,3.1,4.5,3.6,5.2,8.2,7.1,7,9.1]
9  weights = [3.5,5.1,8.6,5.2,8,7,3.5,9,1.9,5.7]
10
11 #neuron 1
12 bias = 3
13
14 #menampilkan keluaran
15 outputs = np.dot(weights, inputs) + bias
16 print(outputs)
```

Output :



The screenshot shows a Jupyter Notebook cell with the following Python code:

```
#Dian Nur Safitri
#21091397044

#inisialisasi library
import numpy as np

#input layer feature 10
inputs = [1,3,2,2.4,6,1.3,5,2.2,1.10,2.2]
weights = [
    [0.5,0.3,-0.5,2.2,0.3,2.2,0.20,-0.18,0.30,0.6],
    [0.8,-0.80,0.26,-0.5,1.0,0.7,0.6,0.30,-0.15,0.3],
    [-0.35,-0.27,0.17,0.87,0.20,1.14,0.71,-0.03,0.29,-1.0],
    [0.21,-0.5,-0.4,0.1,2.03,0.17,0.7,0.10,-2.0,3.2],
    [0.5,0.9,-0.15,0.6,0.50,3.0,-0.15,0.15,-0.28,0.28]
]

#neuron 5
biases = [3,5,0.9,2,0.7]

#menampilkan keluaran
outputs = np.dot (weights, inputs) + biases
print(outputs)
```

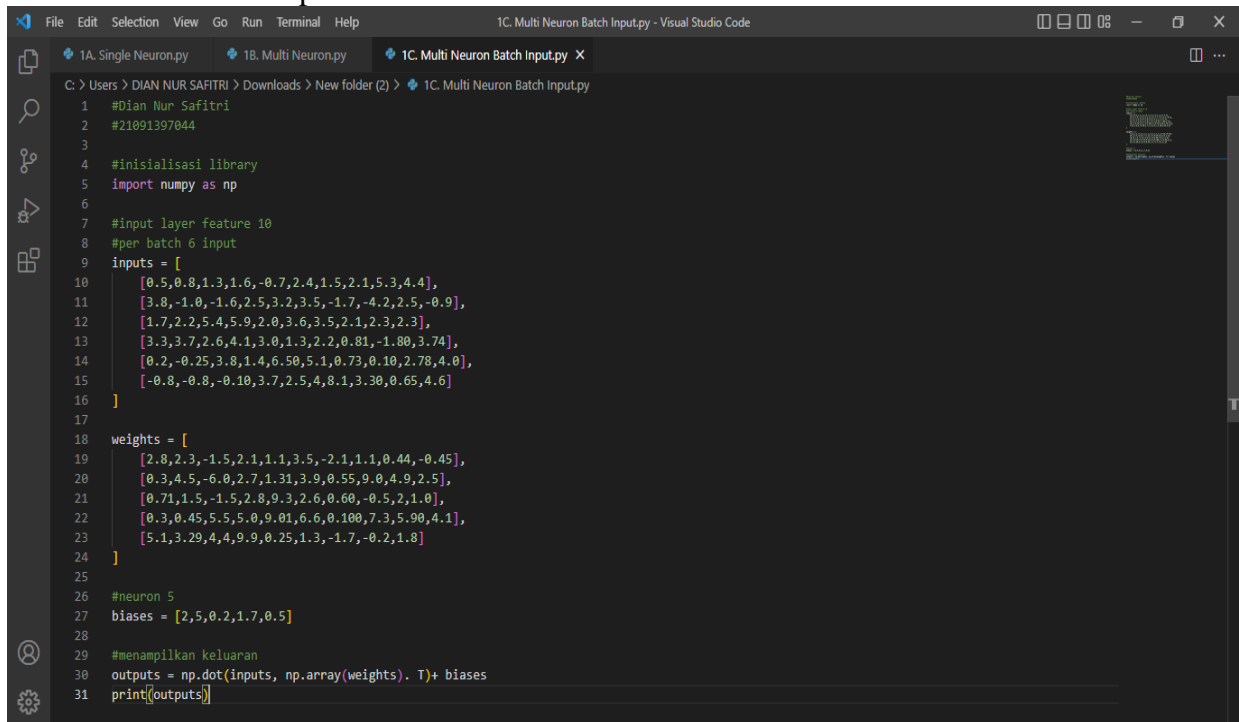
The output of the code is displayed below the cell:

```
[15.594 13.785  6.453 21.111 11.828]
```

Penjelasan Pengerjaan :

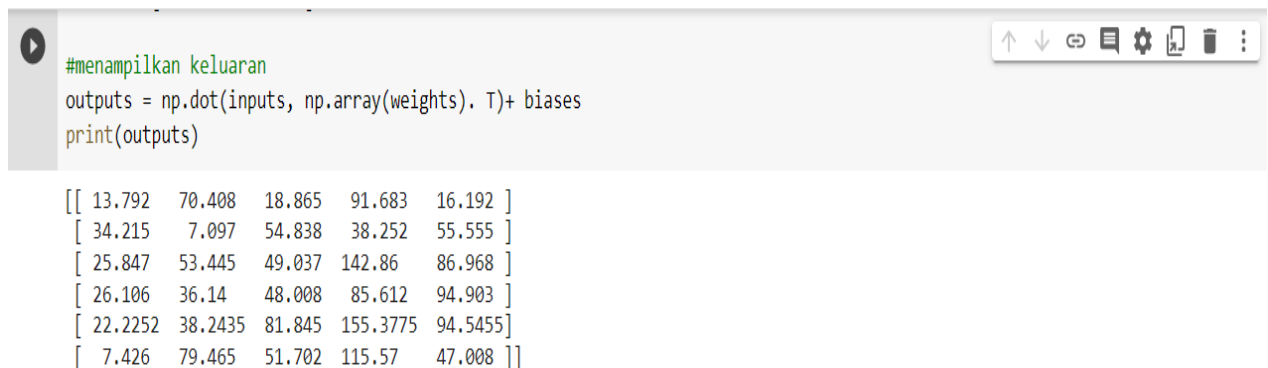
- Import numpy merupakan library python
- Memodelkan 5 neuron dengan 10 input. Lima neuron berarti perlu memasukkan lima set bobot (weights). Dimana setiap set nya memiliki 10 nilai karena ada 10 input
- Penambahan vektor secara digunakan untuk penambahan bias
- Selanjutnya menampilkan keluaran atau neuron. Karena memodelkan lima neuron, outputnya pun akan ada lima, bukan nilai tunggal seperti output single neuron
- Mencetak output, jalankan. print (outputs)
- Hasil output : [15.594 13.785 6.453 21.111 11.828]

3. Multi Neuron Batch Input



```
1  #Dian Nur Safitri
2  #21091397044
3
4  #inisialisasi library
5  import numpy as np
6
7  #input layer feature 10
8  #per batch 6 input
9  inputs = [
10     [0.5,0.8,1.3,1.6,-0.7,2.4,1.5,2.1,5.3,4.4],
11     [3.8,-1.0,-1.6,2.5,3.2,3.5,-1.7,-4.2,2.5,-0.9],
12     [1.7,2.2,5.4,5.9,2.0,3.6,3.5,2.1,2.3,2.3],
13     [3.3,3.7,2.6,4.1,3.0,1.3,2.2,0.81,-1.80,3.74],
14     [0.2,-0.25,3.8,1.4,6.50,5.1,0.73,0.10,2.78,4.0],
15     [-0.8,-0.8,-0.10,3.7,2.5,4.8,1.3,30,0.65,4.6]
16 ]
17
18 weights = [
19     [2.8,2.3,-1.5,2.1,1.1,3.5,-2.1,1.1,0.44,-0.45],
20     [0.3,4.5,-6.0,2.7,1.31,3.9,0.55,9.0,4.9,2.5],
21     [0.71,1.5,-1.5,2.8,9.3,2.6,0.60,-0.5,2.1,0],
22     [0.3,0.45,5.5,5.0,9.01,6.6,0.100,7.3,5.90,4.1],
23     [5.1,3.29,4,4,9.9,0.25,1.3,-1.7,-0.2,1.8]
24 ]
25
26 #neuron 5
27 biases = [2,5,0.2,1.7,0.5]
28
29 #menampilkan keluaran
30 outputs = np.dot(inputs, np.array(weights). T)+ biases
31 print(outputs]
```

Output :



```
#menampilkan keluaran
outputs = np.dot(inputs, np.array(weights). T)+ biases
print(outputs)

[[ 13.792  70.408  18.865  91.683  16.192 ]
 [ 34.215   7.097  54.838  38.252  55.555 ]
 [ 25.847  53.445  49.037 142.86   86.968 ]
 [ 26.106  36.14   48.008  85.612  94.903 ]
 [ 22.2252 38.2435  81.845 155.3775 94.5455]
 [ 7.426   79.465  51.702 115.57   47.008 ]]
```

Penjelasan Pengerjaan :

- Import numpy merupakan library python menambahkan input sebanyak 10 kali
- Memodelkan lima neuron dengan enam input per batch-nya, sehingga menjadi kumpulan input (matriks) yang terdiri dari input 6 set, dan bobot 5 set yang masing-masing set nya memiliki 10 nilai karena ada 10 input
- Perlu melakukan dot product dan menempatkan hasilnya dalam array output
- Lalu tampilkan hasil dengan print (outputs)
- Hasil outputsnya :

```
[ 13.792   70.408   18.865   91.683   16.192 ]  
[ 34.215    7.097   54.838   38.252   55.555 ]  
[ 25.847   53.445   49.037  142.86    86.968 ]  
[ 26.106   36.14    48.008   85.612   94.903 ]  
[ 22.2252  38.2435   81.845  155.3775  94.5455]  
[  7.426   79.465   51.702  115.57    47.008 ]
```

