LAPORAN

UTS KECERDASAN BUATAN PART II

Disusun oleh :

Ananda Putri Rahmadani 21091397046

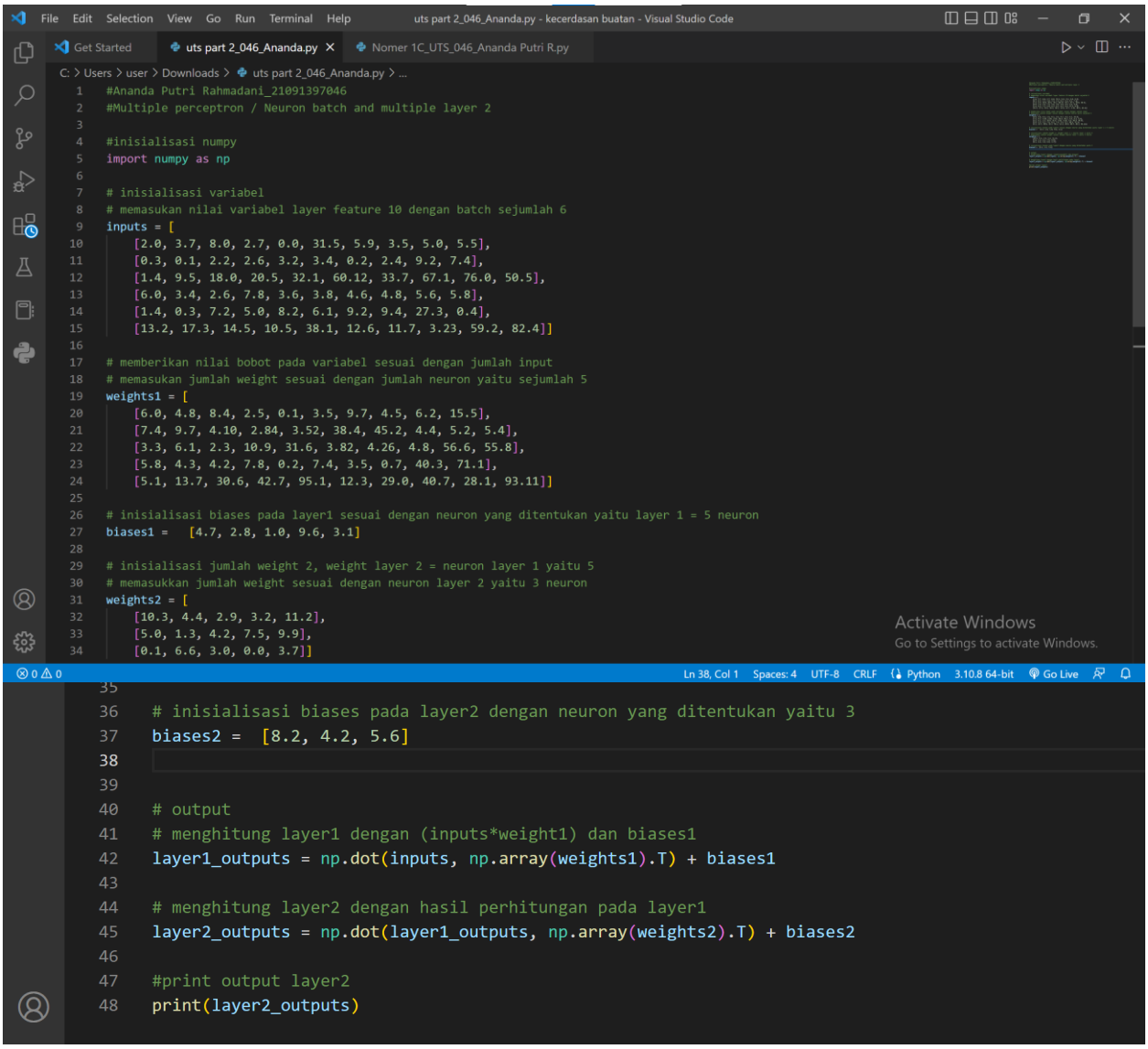
UNIVERSITAS NEGERI SURABAYA

MANAJEMEN INFORMATIKA

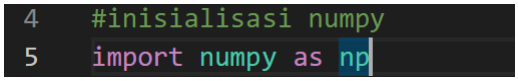
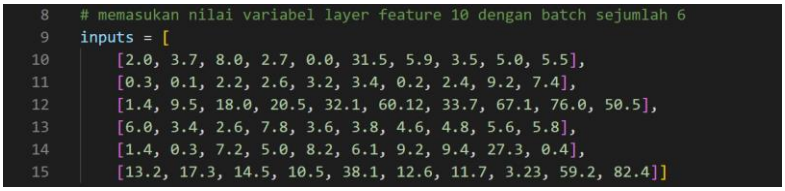
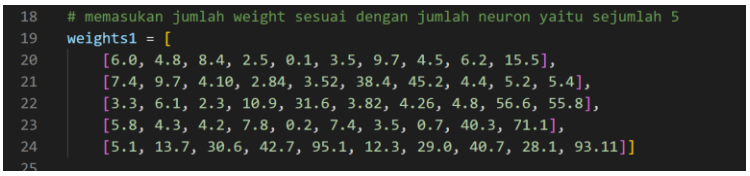
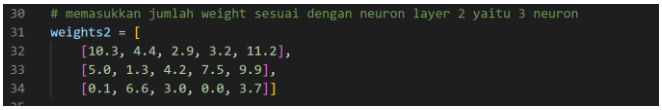
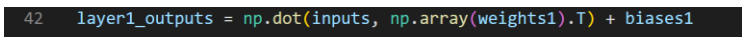
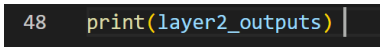
2022

1. Buat kodingan
    a. Multi Neuron Batch Input
        i. Input layer feature 10
        ii. Per batch nya 6 input
        iii. Hidden layer 1, 5 neuron
        iv. Hidden layer 2, 3 neuron

- **Kodingan**



- **Penjelasan**

	Line ke 4 terdapat tanda # ini menunjukan komentar Line ke5 menginisialisasi numpy ke np untuk mempermudah dalam mengoprasionalkan kodingan
	Pada line ke 9 variabel diinisialisasikan , lalu memasukan nilai input dengan jumlah 10 baris angka sesuai dengan yang ditentukan yakni feature layer 10 dan 6 kolom angka sesuai yang diminta yaitu batch = 6
	Line ke 19 memasukan nilai weight dengan 5 kolom , hal ini sesuai dengan neuron yang diperintahkan yaitu 5 dan 10 baris angka
	Line 31 memasukan nilai weight2 dengan jumlah weight sama dengan neuron yang ditentukan yakni 5
	Pada line 42 terdapat rumus untuk menghitung layer 1 dengan weight lalu mentranspose lebih dulu
	Line 45 terdapat rumus untuk menghitung dari hasil layer 1 dikalikan dengan weight transpose lalu dijumlahkan dengan biases 2
	Line 48 ini berfungsi untuk menampilkan hasil output dari kodingan

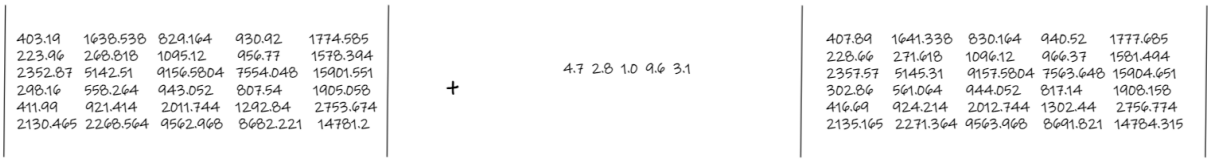
- **Output**

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  JUPYTER

Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS D:\Ananda (semester 3)\kecerdasan buatan> & C:/Users/user/AppData/Local/Programs/Python/Python310/p
.py"
[[ 36758.5658   32317.0597   19947.1463 ]
 [ 27542.382    29008.873    10961.0326 ]
 [275823.28296 271126.19558 120520.3529 ]
 [ 32320.308    31232.2158   13631.249  ]
 [ 49247.283    48803.0156   22385.3772 ]
 [253128.0635  265354.8398   98603.9884 ]]
PS D:\Ananda (semester 3)\kecerdasan buatan>
```

- **Perhitungan Output**

```
41    # menghitung layer1 dengan (inputs*weight1) dan biases1
42    layer1_outputs = np.dot(inputs, np.array(weights1).T) + biases1
```

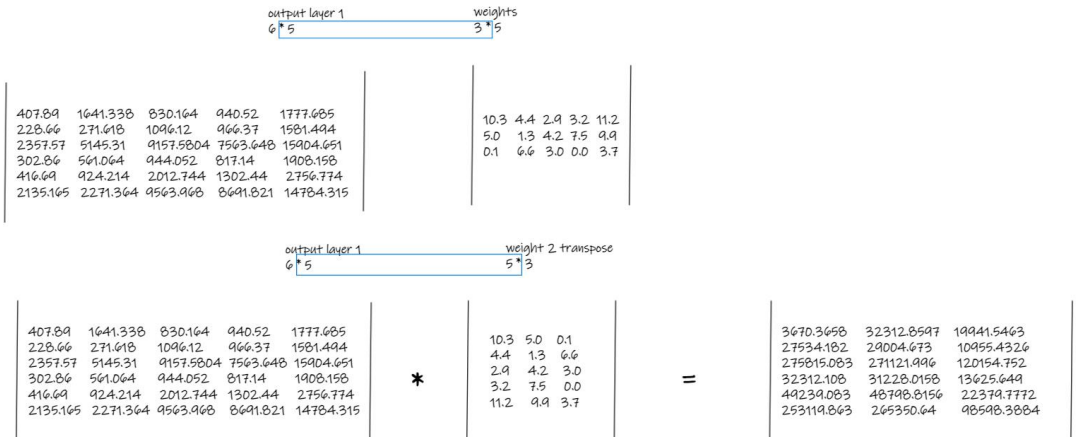
Ketika layer 1sudah ditemukan dengan melakukan perkalian input dan transpose weight1 hasil np.dot ditambah dengan biases 1



Menghitung hidden layer 2

```
43
44    # menghitung layer2 dengan hasil perhitungan pada layer1
45    layer2_outputs = np.dot(layer1_outputs, np.array(weights2).T) + biases2
46
```

Saat hasil dari layer1 didapatkan lalu lanjut masuk ke penghitungan layer 2, mencari nilai np.dot layer 2 dengan cara hasil layer 1 dikali dengan weight transpose 2. Weights 2 di transpose untuk menyamakan ordo output hidden layer 1



Setelah didapatkan hasil np.dot layer 2 lalu ditambah dengan biases 2 dan hasil layer 2 didapatkan