LAPORAN KECERDASAN BUATAN
UJIAN TENGAH SEMESTER

Nama : Cinderella Ih Hsin Chiang

Kelas/NIM : B/21091397022

PRODI D4 MANAJEMEN INFORMATIKA

FAKULTAS VOKASI

UNIVERSITAS NEGERI SURABAYA

```
1. inputs = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
   weights = [0.2, 0.4, 0.6, 0.8, 0.3, 0.5, 0.7, 0.1, 0.9, 0.1]
   bias = 5
   output = (inputs[0]*weights[0] + inputs[1]*weights[1] +
   inputs[2]*weights[2] + inputs[3]*weights[3] + inputs[4]*weights[4] +
   inputs[5]*weights[5] + inputs[6]*weights[6] + inputs[7]*weights[7] +
   inputs[8]*weights[8] + inputs[9]*weights[9] + bias)
   print(output)
   >>>>
   25.69999999999996
```

Pengerjaan : ((0*0.2) + (1*0.4) + (2*0.6) + (3*0.8) + (4*0.3) + (5*0.5) + (6*0.7) + (7*0.1) + (8*0.9) + (9*0.1) + 5) = 25.699999999999996

```
2. inputs = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
   weights = [0.2, 0.4, 0.6, 0.8, 0.3, 0.5, 0.7, 0.1, 0.9, 0.1]
   bias1 = 5
   bias2 = 3
   bias3 = 2
   bias4 = 1
   bias5 = 4
   outputs = [
      (inputs[0]*weights[0] + inputs[1]*weights[1] + inputs[2]*weights[2]
   + inputs[3]*weights[3] +
       inputs[4]*weights[4] + inputs[5]*weights[5] + inputs[6]*weights[6]
   + inputs[7]*weights[7] +
       inputs[8]*weights[8] + inputs[9]*weights[9] + bias1),
      (inputs[0]*weights[0] + inputs[1]*weights[1] + inputs[2]*weights[2]
   + inputs[3]*weights[3] +
       inputs[4]*weights[4] + inputs[5]*weights[5] + inputs[6]*weights[6]
   + inputs[7]*weights[7] +
       inputs[8]*weights[8] + inputs[9]*weights[9] + bias2),
      (inputs[0]*weights[0] + inputs[1]*weights[1] + inputs[2]*weights[2]
   + inputs[3]*weights[3] +
       inputs[4]*weights[4] + inputs[5]*weights[5] + inputs[6]*weights[6]
   + inputs[7]*weights[7] +
       inputs[8]*weights[8] + inputs[9]*weights[9] + bias3),
      (inputs[0]*weights[0] + inputs[1]*weights[1] + inputs[2]*weights[2]
   + inputs[3]*weights[3] +
       inputs[4]*weights[4] + inputs[5]*weights[5] + inputs[6]*weights[6]
   + inputs[7]*weights[7] +
       inputs[8]*weights[8] + inputs[9]*weights[9] + bias4),
      (inputs[0]*weights[0] + inputs[1]*weights[1] + inputs[2]*weights[2]
   + inputs[3]*weights[3] +
       inputs[4]*weights[4] + inputs[5]*weights[5] + inputs[6]*weights[6]
   + inputs[7]*weights[7] +
       inputs[8]*weights[8] + inputs[9]*weights[9] + bias5)]
   print(outputs)

>>>>

[25.69999999999996, 23.69999999999996, 22.69999999999996,
21.69999999999996, 24.69999999999996]
```

Pengerjaan : ((0*0.2) + (1*0.4) + (2*0.6) + (3*0.8) + (4*0.3) + (5*0.5) + (6*0.7) + (7*0.1) + (8*0.9) + (9*0.1) + 5), ((0*0.2) + (1*0.4) + (2*0.6) + (3*0.8) + (4*0.3) + (5*0.5) + (6*0.7) + (7*0.1) + (8*0.9) + (9*0.1) + 3), ((0*0.2) + (1*0.4) + (2*0.6) + (3*0.8) + (4*0.3) + (5*0.5) + (6*0.7) + (7*0.1) + (8*0.9) +

(9*0.1) + 2), ((0*0.2) + (1*0.4) + (2*0.6) + (3*0.8) + (4*0.3) + (5*0.5) + (6*0.7) + (7*0.1) + (8*0.9) +
(9*0.1) + 1), ((0*0.2) + (1*0.4) + (2*0.6) + (3*0.8) + (4*0.3) + (5*0.5) + (6*0.7) + (7*0.1) + (8*0.9) +
(9*0.1) + 4) = 25.699999999999996, 23.699999999999996, 22.699999999999996,
21.699999999999996, 24.699999999999996

```
3. inputs = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
   weights1 = [0.2, 0.4, 0.6, 0.8, 0.3, 0.5, 0.7, 0.1, 0.9, 0.1]
   weights2 = [0.1, 0.2, 0.2, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 0.1]
   weights3 = [0.9, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.6, 0.7, 0.8]
   weights4 = [0.2, 0.2, 0.2, 0.2, 0.2, 0.2, 0.2, 0.2, 0.2, 0.2]
   weights5 = [0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1]
   weights6 = [0.3, 0.3, 0.3, 0.3, 0.3, 0.3, 0.3, 0.3, 0.3, 0.3]
   bias1 = 5
   bias2 = 3
   bias3 = 2
   bias4 = 1
   bias5 = 4
   outputs = [
   # Neuron 1
       (inputs[0]*weights1[0] + inputs[1]*weights1[1] +
   inputs[2]*weights1[2] + inputs[3]*weights1[3] +
           inputs[4]*weights1[4] + inputs[5]*weights1[5] +
   inputs[6]*weights1[6] + inputs[7]*weights1[7] +
           inputs[8]*weights1[8] + inputs[9]*weights1[9] + bias1),
           (inputs[0]*weights2[0] + inputs[1]*weights2[1] +
   inputs[2]*weights2[2] + inputs[3]*weights2[3] +
           inputs[4]*weights2[4] + inputs[5]*weights2[5] +
   inputs[6]*weights2[6] + inputs[7]*weights2[7] +
           inputs[8]*weights2[8] + inputs[9]*weights2[9] + bias1),
           (inputs[0]*weights3[0] + inputs[1]*weights3[1] +
   inputs[2]*weights3[2] + inputs[3]*weights3[3] +
           inputs[4]*weights3[4] + inputs[5]*weights3[5] +
   inputs[6]*weights3[6] + inputs[7]*weights3[7] +
           inputs[8]*weights3[8] + inputs[9]*weights3[9] + bias1),
           (inputs[0]*weights4[0] + inputs[1]*weights4[1] +
   inputs[2]*weights4[2] + inputs[3]*weights4[3] +
           inputs[4]*weights4[4] + inputs[5]*weights4[5] +
   inputs[6]*weights4[6] + inputs[7]*weights4[7] +
           inputs[8]*weights4[8] + inputs[9]*weights4[9] + bias1),
           (inputs[0]*weights5[0] + inputs[1]*weights5[1] +
   inputs[2]*weights5[2] + inputs[3]*weights5[3] +
           inputs[4]*weights5[4] + inputs[5]*weights5[5] +
   inputs[6]*weights5[6] + inputs[7]*weights5[7] +
           inputs[8]*weights5[8] + inputs[9]*weights5[9] + bias1),
           (inputs[0]*weights6[0] + inputs[1]*weights6[1] +
   inputs[2]*weights6[2] + inputs[3]*weights6[3] +
           inputs[4]*weights6[4] + inputs[5]*weights6[5] +
   inputs[6]*weights6[6] + inputs[7]*weights6[7] +
           inputs[8]*weights6[8] + inputs[9]*weights6[9] + bias1),
   # Neuron 2 :
       (inputs[0]*weights1[0] + inputs[1]*weights1[1] +
   inputs[2]*weights1[2] + inputs[3]*weights1[3] +
           inputs[4]*weights1[4] + inputs[5]*weights1[5] +
   inputs[6]*weights1[6] + inputs[7]*weights1[7] +
           inputs[8]*weights1[8] + inputs[9]*weights1[9] + bias2),
           (inputs[0]*weights2[0] + inputs[1]*weights2[1] +
```

```
          inputs[2]*weights2[2] + inputs[3]*weights2[3] +
          inputs[4]*weights2[4] + inputs[5]*weights2[5] +
inputs[6]*weights2[6] + inputs[7]*weights2[7] +
          inputs[8]*weights2[8] + inputs[9]*weights2[9] + bias2),
          (inputs[0]*weights3[0] + inputs[1]*weights3[1] +
inputs[2]*weights3[2] + inputs[3]*weights3[3] +
          inputs[4]*weights3[4] + inputs[5]*weights3[5] +
inputs[6]*weights3[6] + inputs[7]*weights3[7] +
          inputs[8]*weights3[8] + inputs[9]*weights3[9] + bias2),
          (inputs[0]*weights4[0] + inputs[1]*weights4[1] +
inputs[2]*weights4[2] + inputs[3]*weights4[3] +
          inputs[4]*weights4[4] + inputs[5]*weights4[5] +
inputs[6]*weights4[6] + inputs[7]*weights4[7] +
          inputs[8]*weights4[8] + inputs[9]*weights4[9] + bias2),
          (inputs[0]*weights5[0] + inputs[1]*weights5[1] +
inputs[2]*weights5[2] + inputs[3]*weights5[3] +
          inputs[4]*weights5[4] + inputs[5]*weights5[5] +
inputs[6]*weights5[6] + inputs[7]*weights5[7] +
          inputs[8]*weights5[8] + inputs[9]*weights5[9] + bias2),
          (inputs[0]*weights6[0] + inputs[1]*weights6[1] +
inputs[2]*weights6[2] + inputs[3]*weights6[3] +
          inputs[4]*weights6[4] + inputs[5]*weights6[5] +
inputs[6]*weights6[6] + inputs[7]*weights6[7] +
          inputs[8]*weights6[8] + inputs[9]*weights6[9] + bias2),
# Neuron 3 :
   (inputs[0]*weights1[0] + inputs[1]*weights1[1] +
inputs[2]*weights1[2] + inputs[3]*weights1[3] +
          inputs[4]*weights1[4] + inputs[5]*weights1[5] +
inputs[6]*weights1[6] + inputs[7]*weights1[7] +
          inputs[8]*weights1[8] + inputs[9]*weights1[9] + bias3),
          (inputs[0]*weights2[0] + inputs[1]*weights2[1] +
inputs[2]*weights2[2] + inputs[3]*weights2[3] +
          inputs[4]*weights2[4] + inputs[5]*weights2[5] +
inputs[6]*weights2[6] + inputs[7]*weights2[7] +
          inputs[8]*weights2[8] + inputs[9]*weights2[9] + bias3),
          (inputs[0]*weights3[0] + inputs[1]*weights3[1] +
inputs[2]*weights3[2] + inputs[3]*weights3[3] +
          inputs[4]*weights3[4] + inputs[5]*weights3[5] +
inputs[6]*weights3[6] + inputs[7]*weights3[7] +
          inputs[8]*weights3[8] + inputs[9]*weights3[9] + bias3),
          (inputs[0]*weights4[0] + inputs[1]*weights4[1] +
inputs[2]*weights4[2] + inputs[3]*weights4[3] +
          inputs[4]*weights4[4] + inputs[5]*weights4[5] +
inputs[6]*weights4[6] + inputs[7]*weights4[7] +
          inputs[8]*weights4[8] + inputs[9]*weights4[9] + bias3),
          (inputs[0]*weights5[0] + inputs[1]*weights5[1] +
inputs[2]*weights5[2] + inputs[3]*weights5[3] +
          inputs[4]*weights5[4] + inputs[5]*weights5[5] +
inputs[6]*weights5[6] + inputs[7]*weights5[7] +
          inputs[8]*weights5[8] + inputs[9]*weights5[9] + bias3),
          (inputs[0]*weights6[0] + inputs[1]*weights6[1] +
inputs[2]*weights6[2] + inputs[3]*weights6[3] +
          inputs[4]*weights6[4] + inputs[5]*weights6[5] +
inputs[6]*weights6[6] + inputs[7]*weights6[7] +
          inputs[8]*weights6[8] + inputs[9]*weights6[9] + bias3),
# Neuron 4 :
   (inputs[0]*weights1[0] + inputs[1]*weights1[1] +
```

```
    inputs[2]*weights1[2] + inputs[3]*weights1[3] +
        inputs[4]*weights1[4] + inputs[5]*weights1[5] +
inputs[6]*weights1[6] + inputs[7]*weights1[7] +
        inputs[8]*weights1[8] + inputs[9]*weights1[9] + bias4),
        (inputs[0]*weights2[0] + inputs[1]*weights2[1] +
inputs[2]*weights2[2] + inputs[3]*weights2[3] +
        inputs[4]*weights2[4] + inputs[5]*weights2[5] +
inputs[6]*weights2[6] + inputs[7]*weights2[7] +
        inputs[8]*weights2[8] + inputs[9]*weights2[9] + bias4),
        (inputs[0]*weights3[0] + inputs[1]*weights3[1] +
inputs[2]*weights3[2] + inputs[3]*weights3[3] +
        inputs[4]*weights3[4] + inputs[5]*weights3[5] +
inputs[6]*weights3[6] + inputs[7]*weights3[7] +
        inputs[8]*weights3[8] + inputs[9]*weights3[9] + bias4),
        (inputs[0]*weights4[0] + inputs[1]*weights4[1] +
inputs[2]*weights4[2] + inputs[3]*weights4[3] +
        inputs[4]*weights4[4] + inputs[5]*weights4[5] +
inputs[6]*weights4[6] + inputs[7]*weights4[7] +
        inputs[8]*weights4[8] + inputs[9]*weights4[9] + bias4),
        (inputs[0]*weights5[0] + inputs[1]*weights5[1] +
inputs[2]*weights5[2] + inputs[3]*weights5[3] +
        inputs[4]*weights5[4] + inputs[5]*weights5[5] +
inputs[6]*weights5[6] + inputs[7]*weights5[7] +
        inputs[8]*weights5[8] + inputs[9]*weights5[9] + bias4),
        (inputs[0]*weights6[0] + inputs[1]*weights6[1] +
inputs[2]*weights6[2] + inputs[3]*weights6[3] +
        inputs[4]*weights6[4] + inputs[5]*weights6[5] +
inputs[6]*weights6[6] + inputs[7]*weights6[7] +
        inputs[8]*weights6[8] + inputs[9]*weights6[9] + bias4),
# Neuron 5 :
    (inputs[0]*weights1[0] + inputs[1]*weights1[1] +
inputs[2]*weights1[2] + inputs[3]*weights1[3] +
        inputs[4]*weights1[4] + inputs[5]*weights1[5] +
inputs[6]*weights1[6] + inputs[7]*weights1[7] +
        inputs[8]*weights1[8] + inputs[9]*weights1[9] + bias5),
        (inputs[0]*weights2[0] + inputs[1]*weights2[1] +
inputs[2]*weights2[2] + inputs[3]*weights2[3] +
        inputs[4]*weights2[4] + inputs[5]*weights2[5] +
inputs[6]*weights2[6] + inputs[7]*weights2[7] +
        inputs[8]*weights2[8] + inputs[9]*weights2[9] + bias5),
        (inputs[0]*weights3[0] + inputs[1]*weights3[1] +
inputs[2]*weights3[2] + inputs[3]*weights3[3] +
        inputs[4]*weights3[4] + inputs[5]*weights3[5] +
inputs[6]*weights3[6] + inputs[7]*weights3[7] +
        inputs[8]*weights3[8] + inputs[9]*weights3[9] + bias5),
        (inputs[0]*weights4[0] + inputs[1]*weights4[1] +
inputs[2]*weights4[2] + inputs[3]*weights4[3] +
        inputs[4]*weights4[4] + inputs[5]*weights4[5] +
inputs[6]*weights4[6] + inputs[7]*weights4[7] +
        inputs[8]*weights4[8] + inputs[9]*weights4[9] + bias5),
        (inputs[0]*weights5[0] + inputs[1]*weights5[1] +
inputs[2]*weights5[2] + inputs[3]*weights5[3] +
        inputs[4]*weights5[4] + inputs[5]*weights5[5] +
inputs[6]*weights5[6] + inputs[7]*weights5[7] +
        inputs[8]*weights5[8] + inputs[9]*weights5[9] + bias5),
        (inputs[0]*weights6[0] + inputs[1]*weights6[1] +
inputs[2]*weights6[2] + inputs[3]*weights6[3] +
```

```
          inputs[4]*weights6[4] + inputs[5]*weights6[5] +
    inputs[6]*weights6[6] + inputs[7]*weights6[7] +
          inputs[8]*weights6[8] + inputs[9]*weights6[9] + bias5)]
    print(outputs)
    >>>>
    [25.69999999999996, 29.7, 31.099999999999998, 14.000000000000002, 9.5,
    18.5, 23.699999999999996, 27.7, 29.099999999999998, 12.000000000000002,
    7.500000000000001, 16.5, 22.699999999999996, 26.7, 28.099999999999998,
    11.000000000000002, 6.500000000000001, 15.5, 21.699999999999996, 25.7,
    27.099999999999998, 10.000000000000002, 5.500000000000001, 14.5,
    24.699999999999996, 28.7, 30.099999999999998, 13.000000000000002, 8.5,
    17.5]
```

Pengerjaan :

Neuron 1 : ((0*0.2) + (1*0.4) + (2*0.6) + (3*0.8) + (4*0.3) + (5*0.5) + (6*0.7) + (7*0.1) + (8*0.9) + (9*0.1) + 5), ((0*0.1) + (1*0.2) + (2*0.2) + (3*0.4) + (4*0.5) + (5*0.6) + (6*0.7) + (7*0.8) + (8*0.9) + (9*0.1) + 5), ((0*0.9) + (1*0.1) + (2*0.2) + (3*0.3) + (4*0.4) + (5*0.5) + (6*0.6) + (7*0.6) + (8*0.7) + (9*0.8) + 5), ((0*0.2) + (1*0.2) + (2*0.2) + (3*0.2) + (4*0.2) + (5*0.2) + (6*0.2) + (7*0.2) + (8*0.2) + (9*0.2) + 5), ((0*0.1) + (1*0.1) + (2*0.1) + (3*0.1) + (4*0.1) + (5*0.1) + (6*0.1) + (7*0.1) + (8*0.1) + (9*0.1) + 5), ((0*0.3) + (1*0.3) + (2*0.3) + (3*0.3) + (4*0.3) + (5*0.3) + (6*0.3) + (7*0.3) + (8*0.3) + (9*0.3) + 5) = 25.69999999999996, 29.7, 31.099999999999998, 14.000000000000002, 9.5, 18.5

Neuron 2 : ((0*0.2) + (1*0.4) + (2*0.6) + (3*0.8) + (4*0.3) + (5*0.5) + (6*0.7) + (7*0.1) + (8*0.9) + (9*0.1) + 3), ((0*0.1) + (1*0.2) + (2*0.2) + (3*0.4) + (4*0.5) + (5*0.6) + (6*0.7) + (7*0.8) + (8*0.9) + (9*0.1) + 3), ((0*0.9) + (1*0.1) + (2*0.2) + (3*0.3) + (4*0.4) + (5*0.5) + (6*0.6) + (7*0.6) + (8*0.7) + (9*0.8) + 3), ((0*0.2) + (1*0.2) + (2*0.2) + (3*0.2) + (4*0.2) + (5*0.2) + (6*0.2) + (7*0.2) + (8*0.2) + (9*0.2) + 3), ((0*0.1) + (1*0.1) + (2*0.1) + (3*0.1) + (4*0.1) + (5*0.1) + (6*0.1) + (7*0.1) + (8*0.1) + (9*0.1) + 3), ((0*0.3) + (1*0.3) + (2*0.3) + (3*0.3) + (4*0.3) + (5*0.3) + (6*0.3) + (7*0.3) + (8*0.3) + (9*0.3) + 3) = 23.69999999999996, 27.7, 29.099999999999998, 12.000000000000002, 7.500000000000001, 16.5

Neuron 3 : ((0*0.2) + (1*0.4) + (2*0.6) + (3*0.8) + (4*0.3) + (5*0.5) + (6*0.7) + (7*0.1) + (8*0.9) + (9*0.1) + 2), ((0*0.1) + (1*0.2) + (2*0.2) + (3*0.4) + (4*0.5) + (5*0.6) + (6*0.7) + (7*0.8) + (8*0.9) + (9*0.1) + 2), ((0*0.9) + (1*0.1) + (2*0.2) + (3*0.3) + (4*0.4) + (5*0.5) + (6*0.6) + (7*0.6) + (8*0.7) + (9*0.8) + 2), ((0*0.2) + (1*0.2) + (2*0.2) + (3*0.2) + (4*0.2) + (5*0.2) + (6*0.2) + (7*0.2) + (8*0.2) + (9*0.2) + 2), ((0*0.1) + (1*0.1) + (2*0.1) + (3*0.1) + (4*0.1) + (5*0.1) + (6*0.1) + (7*0.1) + (8*0.1) + (9*0.1) + 2), ((0*0.3) + (1*0.3) + (2*0.3) + (3*0.3) + (4*0.3) + (5*0.3) + (6*0.3) + (7*0.3) + (8*0.3) + (9*0.3) + 2) = 22.69999999999996, 26.7, 28.099999999999998, 11.000000000000002, 6.500000000000001, 15.5

Neuron 4 : ((0*0.2) + (1*0.4) + (2*0.6) + (3*0.8) + (4*0.3) + (5*0.5) + (6*0.7) + (7*0.1) + (8*0.9) + (9*0.1) + 1), ((0*0.1) + (1*0.2) + (2*0.2) + (3*0.4) + (4*0.5) + (5*0.6) + (6*0.7) + (7*0.8) + (8*0.9) + (9*0.1) + 1), ((0*0.9) + (1*0.1) + (2*0.2) + (3*0.3) + (4*0.4) + (5*0.5) + (6*0.6) + (7*0.6) + (8*0.7) + (9*0.8) + 1), ((0*0.2) + (1*0.2) + (2*0.2) + (3*0.2) + (4*0.2) + (5*0.2) + (6*0.2) + (7*0.2) + (8*0.2) + (9*0.2) + 1), ((0*0.1) + (1*0.1) + (2*0.1) + (3*0.1) + (4*0.1) + (5*0.1) + (6*0.1) + (7*0.1) + (8*0.1) + (9*0.1) + 1), ((0*0.3) + (1*0.3) + (2*0.3) + (3*0.3) + (4*0.3) + (5*0.3) + (6*0.3) + (7*0.3) + (8*0.3) + (9*0.3) + 1) = 21.69999999999996, 25.7, 27.099999999999998, 10.000000000000002, 5.500000000000001, 14.5

Neuron 5 : ((0*0.2) + (1*0.4) + (2*0.6) + (3*0.8) + (4*0.3) + (5*0.5) + (6*0.7) + (7*0.1) + (8*0.9) + (9*0.1) + 4), ((0*0.1) + (1*0.2) + (2*0.2) + (3*0.4) + (4*0.5) + (5*0.6) + (6*0.7) + (7*0.8) + (8*0.9) + (9*0.1) + 4),

((0*0.9) + (1*0.1) + (2*0.2) + (3*0.3) + (4*0.4) + (5*0.5) + (6*0.6) + (7*0.6) + (8*0.7) + (9*0.8) + 4),
((0*0.2) + (1*0.2) + (2*0.2) + (3*0.2) + (4*0.2) + (5*0.2) + (6*0.2) + (7*0.2) + (8*0.2) + (9*0.2) + 4),
((0*0.1) + (1*0.1) + (2*0.1) + (3*0.1) + (4*0.1) + (5*0.1) + (6*0.1) + (7*0.1) + (8*0.1) + (9*0.1) + 4),
((0*0.3) + (1*0.3) + (2*0.3) + (3*0.3) + (4*0.3) + (5*0.3) + (6*0.3) + (7*0.3) + (8*0.3) + (9*0.3) + 4) =
24.699999999999996, 28.7, 30.099999999999998, 13.000000000000002, 8.5, 17.5

```
4. import numpy as np
   inputs = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
   weights = [0.2, 0.4, 0.6, 0.8, 0.3, 0.5, 0.7, 0.1, 0.9, 0.1]
   bias = 5
   output = np.dot(weights, inputs) + bias
   print(output)
   >>>>
   25.7
```

```
5. import numpy as np
   inputs = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
   weights = [0.2, 0.4, 0.6, 0.8, 0.3, 0.5, 0.7, 0.1, 0.9, 0.1]
   biases = [5, 3, 2, 1, 4]
   layer_outputs = np.dot(weights, inputs) + biases
   print(layer_outputs)
   >>>>
   [25.7 23.7 22.7 21.7 24.7]
```

```
6. import numpy as np
   inputs = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
   weights1 = [0.2, 0.4, 0.6, 0.8, 0.3, 0.5, 0.7, 0.1, 0.9, 0.1]
   weights2 = [0.1, 0.2, 0.2, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 0.1]
   weights3 = [0.9, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.6, 0.7, 0.8]
   weights4 = [0.2, 0.2, 0.2, 0.2, 0.2, 0.2, 0.2, 0.2, 0.2, 0.2]
   weights5 = [0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1]
   weights6 = [0.3, 0.3, 0.3, 0.3, 0.3, 0.3, 0.3, 0.3, 0.3, 0.3]
   biases = [5, 3, 2, 1, 4]
   layer_outputs = np.dot(weights1, inputs) + biases, np.dot(weights2,
   inputs) + biases, np.dot(weights3, inputs) + biases,
   np.dot(weights4, inputs) + biases, np.dot(weights5, inputs) + biases,
   np.dot(weights6, inputs) + biases
   print(layer_outputs)
   >>>>
   (array([25.7, 23.7, 22.7, 21.7, 24.7]), array([29.7, 27.7, 26.7, 25.7,
   28.7]), array([31.1, 29.1, 28.1, 27.1, 30.1]))
```

Transpose :

$$
\begin{bmatrix}
25.7 & 23.7 & 22.7 & 21.7 & 24.7 \\
29.7 & 27.7 & 26.7 & 25.7 & 28.7 \\
31.1 & 29.1 & 28.1 & 27.1 & 30.1
\end{bmatrix}
=
\begin{bmatrix}
25.7 & 29.7 & 31.1 \\
23.7 & 27.7 & 29.1 \\
22.7 & 26.7 & 28.1 \\
21.7 & 25.7 & 27.1 \\
24.7 & 28.7 & 30.1
\end{bmatrix}
$$