Student_ID: 19652
Student_Name: Shuilian Shen

21. Project: Design XOR Gate
    - Step 1: Study the general idea on how to design XOR Gate
    - Step 2: Using the following rules to design your own AND Gate, OR Gate, and NAND Gate
        - The forward/backward process

```
- Forward process
  Calculate the output Z for the given input (X,Y).
- Backward process
  Adjust weights
  + If the output Z is too low, increase the weights by 0.5
    which had inputs that were "1".
  + If the output Z is too high, decrease the weights by 0.5
    which had inputs that were "1".
```

Step 2:

AND Gate

```
Desired
"And"
Function

X Y | Z
--------
0 0 | 0
0 1 | 0
1 0 | 0
1 1 | 1
############################
Loop 1
W1=W2=0
Function
X Y | Z
--------
0 0 | 0
0 1 | 0
1 0 | 0
1 1 | 0
```

```
Loop 2
W1=W2=0.5
Function
X Y | Z
--------
0 0 | 0
0 1 | 0
1 0 | 0
1 1 | 1
```

OR Gate

Desired
"OR"
Function
X Y | Z
----
0 0 | 0
0 1 | 1
1 0 | 1
1 1 | 1
############################
Loop 1
W1=W2=0
Function
X Y | Z
----
0 0 | 0
0 1 | 0
1 0 | 0
1 1 | 0

Loop 2
W1=W2=0.5
Function
X Y | Z
----
0 0 | 0
0 1 | 0
1 0 | 0
1 1 | 1

Loop 3
W1=W2=1
Function
X Y | Z
----
0 0 | 0
0 1 | 1
1 0 | 1
1 1 | 1

NAND Gate

Train NOR gate to get W1, W2, Y

Z := (W0 * C + W1 * X + W2 * Y >= T)
   where T := 1.0.

Desired "NOR" Function	Loop 1 W0=0.0 W1=W2=0.5 Function	Loop 2 W0=0.5 W1=W2=0.5 Function	Loop 3 W0=1 W1=W2=0.5 Function
C X Y | Z	C X Y | Z	C X Y | Z	C X Y | Z
1 0 0 | 1	1 0 0 | 0	1 0 0 | 0	1 0 0 | 1
1 0 1 | 1	1 0 1 | 0	1 0 1 | 1	1 0 1 | 1
1 1 0 | 1	1 1 0 | 0	1 1 0 | 1	1 1 0 | 1
1 1 1 | 0	1 1 1 | 1	1 1 1 | 1	1 1 1 | 1

Loop 4 W0=1 W1=W2=0.0 Function	Loop 5 W0=1 W1=W2=-0.5 Function	Loop 6 W0=1.5 W1=W2=-0.5 Function
C X Y | Z	C X Y | Z	C X Y | Z
1 0 0 | 1	1 0 0 | 1	1 0 0 | 1
1 0 1 | 1	1 0 1 | 0	1 0 1 | 1
1 1 0 | 1	1 1 0 | 0	1 1 0 | 1
1 1 1 | 1	1 1 1 | 0	1 1 1 | 0

Step 3:

Z1 := X "AND" Y
=0.5 * X + 0.5 * Y >= 1.0

Z1 := X "OR" Y
=1 * X + 1 * Y >= 1.0

Z2 := X "NAND" Y
=1.5 + -0.5 * X + -0.5 * Y >= 1.0

Z := Z3 := Z1 "AND" Z2
=(1 * X + 1 * Y >= 1.0) "AND" (1.5 + -0.5 * X + -0.5 * Y >= 1.0)

Z:= (0.5 * (1.0 * X + 1.0 * Y >= 1.0) + 0.5 * (1.5 * 1.0 + -0.5 * X + -0.5 * Y >= 1.0)

Step 4:

Please prove that the neural network works for XOR Gate

X=1, Y=1

```
Z :=  (0.5 * ( 1.0 * 1.0 + 1.0 * 1.0 >= 1.0 ) +
      0.5 * ( 1.5 + -0.5 * 1.0 + -0.5 * 1.0 >= 1.0 >= 1.0 )
Z := ( 0.5 + 0.0 >= 1.0 )
Z := ( false )
Z := 0
```

X=1, Y=0

```
Z :=  (0.5 * ( 1.0 * 1.0 + 1.0 * 0.0 >= 1.0 ) +
      0.5 * ( 1.5 + -0.5 * 1.0 + -0.5 * 0.0 >= 1.0 >= 1.0 )
Z := ( 0.5 + 0.5 >= 1.0 )
Z := ( true  )
Z := 1
```

X=0, Y=1

```
Z :=  (0.5 * ( 1.0 * 0.0 + 1.0 * 1.0 >= 1.0 ) +
      0.5 * ( 1.5 + -0.5 * 0.0 + -0.5 * 1.0 >= 1.0 >= 1.0 )
Z := ( 0.5 + 0.5 >= 1.0 )
Z := ( true  )
Z := 1
```

X=0, Y=0

```
Z :=  (0.5 * ( 1.0 * 0.0 + 1.0 * 0.0 >= 1.0 ) +
      0.5 * ( 1.5 + -0.5 * 0.0 + -0.5 * 0.0 >= 1.0 >= 1.0 )
Z := ( 0.0 + 0.5 >= 1.0 )
Z := ( false )
Z := 0
```

Hence, proved that OR AND NAND = XOR