

## Program no.2

## Solving XOR problem using deep Feed Forward Neural Network

```

import numpy as np
from keras.layers import Dense
from keras.models import Sequential

model=Sequential()
model.add(Dense(units=2,activation='relu',input_dim=2))
model.add(Dense(units=1,activation='sigmoid'))
model.compile(loss='binary_crossentropy',optimizer='adam',metrics=['accuracy'])

X=np.array([[0.,0.],[0.,1.],[1.,0.],[1.,1.]])
Y=np.array([0.,1.,1.,0.])
model.fit(X,Y,epochs=1000,batch_size=4)
print(model.get_weights())
print(model.predict(X,batch_size=4))

1/1 [=====] - 0s 7ms/step - loss: 0.5841 - accuracy: 0.7500
Epoch 972/1000
1/1 [=====] - 0s 7ms/step - loss: 0.5840 - accuracy: 0.7500
Epoch 973/1000
1/1 [=====] - 0s 11ms/step - loss: 0.5839 - accuracy: 0.7500
Epoch 974/1000
1/1 [=====] - 0s 8ms/step - loss: 0.5838 - accuracy: 0.7500
Epoch 975/1000
1/1 [=====] - 0s 7ms/step - loss: 0.5836 - accuracy: 0.7500
Epoch 976/1000
1/1 [=====] - 0s 7ms/step - loss: 0.5835 - accuracy: 0.7500
Epoch 977/1000
1/1 [=====] - 0s 7ms/step - loss: 0.5833 - accuracy: 0.7500
Epoch 978/1000
1/1 [=====] - 0s 7ms/step - loss: 0.5832 - accuracy: 0.7500
Epoch 979/1000
1/1 [=====] - 0s 7ms/step - loss: 0.5831 - accuracy: 0.7500
Epoch 980/1000
1/1 [=====] - 0s 6ms/step - loss: 0.5829 - accuracy: 0.7500
Epoch 981/1000
1/1 [=====] - 0s 6ms/step - loss: 0.5828 - accuracy: 0.7500
Epoch 982/1000
1/1 [=====] - 0s 6ms/step - loss: 0.5827 - accuracy: 0.7500
Epoch 983/1000
1/1 [=====] - 0s 7ms/step - loss: 0.5825 - accuracy: 0.7500
Epoch 984/1000
1/1 [=====] - 0s 7ms/step - loss: 0.5824 - accuracy: 0.7500
Epoch 985/1000
1/1 [=====] - 0s 7ms/step - loss: 0.5823 - accuracy: 0.7500
Epoch 986/1000
1/1 [=====] - 0s 7ms/step - loss: 0.5821 - accuracy: 0.7500
Epoch 987/1000
1/1 [=====] - 0s 7ms/step - loss: 0.5820 - accuracy: 0.7500
Epoch 988/1000
1/1 [=====] - 0s 6ms/step - loss: 0.5819 - accuracy: 0.7500

```

```
Epoch 989/1000
1/1 [=====] - 0s 7ms/step - loss: 0.5817 - accuracy: 0.750
Epoch 990/1000
1/1 [=====] - 0s 6ms/step - loss: 0.5816 - accuracy: 0.750
Epoch 991/1000
1/1 [=====] - 0s 6ms/step - loss: 0.5815 - accuracy: 0.750
Epoch 992/1000
1/1 [=====] - 0s 7ms/step - loss: 0.5813 - accuracy: 0.750
Epoch 993/1000
1/1 [=====] - 0s 7ms/step - loss: 0.5812 - accuracy: 0.750
Epoch 994/1000
1/1 [=====] - 0s 7ms/step - loss: 0.5811 - accuracy: 0.750
Epoch 995/1000
1/1 [=====] - 0s 7ms/step - loss: 0.5810 - accuracy: 0.750
Epoch 996/1000
1/1 [=====] - 0s 7ms/step - loss: 0.5808 - accuracy: 0.750
Epoch 997/1000
1/1 [=====] - 0s 7ms/step - loss: 0.5807 - accuracy: 0.750
Epoch 998/1000
1/1 [=====] - 0s 7ms/step - loss: 0.5806 - accuracy: 0.750
Epoch 999/1000
1/1 [=====] - 0s 7ms/step - loss: 0.5805 - accuracy: 0.750
Epoch 1000/1000
```

✓ 24s completed at 00:45

● ✕