PROGRAM-18

FEED FORWARD NEURAL NETWORK PROBLEM

AIM:-

To write and execute the python program for the Feed Forward Neural Network program.

PROCEDURE:-

Import Required Libraries:

- Import the necessary module numpy

Initialize Parameters:

- Define a function to initialize the parameters of the neural network.
- Initialize weights and biases for the input-hidden and hidden-output layers.

Forward Propagation:

- Define a function to perform forward propagation in the neural network.
- Calculate the output of the hidden layer using the sigmoid activation function.
- Calculate the predicted output using the output layer.
- Return the predicted output.

Execute Forward Propagation:

- Perform forward propagation using the initialized parameters and input data.
- Print the predicted output.

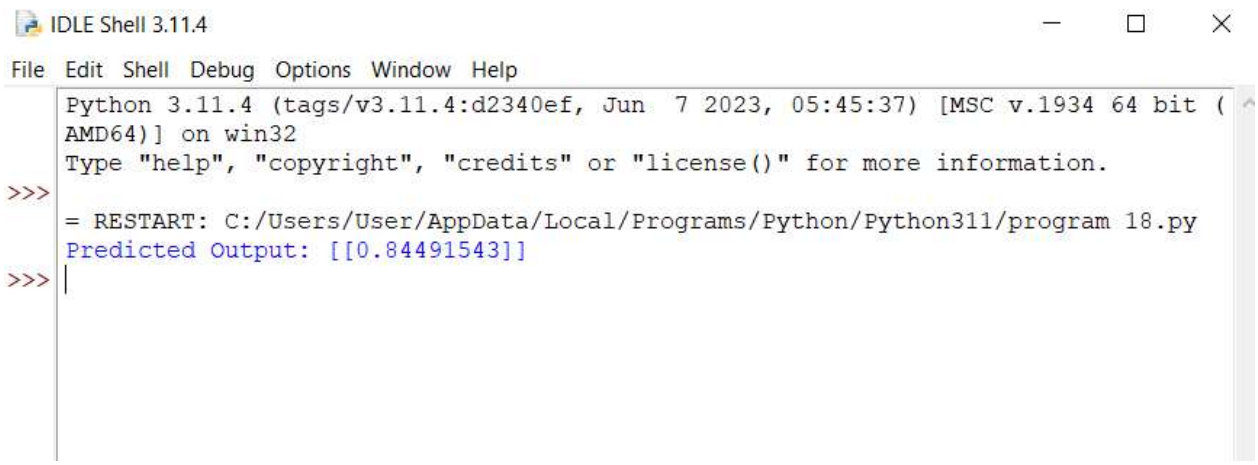
.

CODING:-

```
import numpy as np

def sigmoid(x):

    return 1 / (1 + np.exp(-x))

def initialize_parameters(input_size, hidden_size, output_size):

    return np.random.rand(input_size, hidden_size), np.zeros((1, hidden_size)),
np.random.rand(hidden_size, output_size), np.zeros((1, output_size))

def forward(X, weights_input_hidden, bias_hidden, weights_hidden_output,
bias_output):
```

```
    hidden_layer_output = sigmoid(np.dot(X, weights_input_hidden) + bias_hidden)

    predicted_output = sigmoid(np.dot(hidden_layer_output, weights_hidden_output) +
bias_output)

    return predicted_output

input_size, hidden_size, output_size = 3, 4, 1

weights_input_hidden, bias_hidden, weights_hidden_output, bias_output =
initialize_parameters(input_size, hidden_size, output_size)

input_data = np.array([[0.2, 0.3, 0.4]])

predicted_output = forward(input_data, weights_input_hidden, bias_hidden,
weights_hidden_output, bias_output)

print("Predicted Output:", predicted_output)
```

OUTPUT:-



RESULT:-

Hence the program has been successfully executed and verified.