```java
import java.util.ArrayList;
public class Neuron {
    private float activationValue, weight;
    private String neuronName;
    private ArrayList<Neuron> outputs = new ArrayList<Neuron>();
    private float[] inputs;
    int inputCounter = 0, nInputs;

    public Neuron(float activationValue, float weight, String neuronName, int nInputs){
        this.activationValue = activationValue;
        this.weight = weight;
        this.neuronName = neuronName;
        this.nInputs = nInputs;
        inputs = new float[nInputs];
    }
    public void connect(Neuron neuron){
        outputs.add(neuron);
    }

    public void input(float inputValue){
        inputs[inputCounter] = inputValue;
        inputCounter++;
        if(inputCounter == nInputs){
            fire();
        }
    }

    public void fire(){
        float sum = 0;
        for(int i = 0; i < nInputs; i++){
            sum+=inputs[i];
        }

        float signal = sum*weight;


        if(signal > activationValue){
            for(int i = 0; i < outputs.size(); i++){
                outputs.get(i).input(signal);
            }
        } else{
            for(int i = 0; i < outputs.size(); i++){
                outputs.get(i).input(0);
            }
        }
        System.out.println(neuronName + ":" + signal);
    }
public static void main(String[] args) {
    Neuron inputNode = new Neuron(0,1,"InputNode",1);
    Neuron hiddenNode_One = new Neuron(0.5f,0.9f,"HiddenNode_One",1);
    Neuron hiddenNode_Two = new Neuron(0.5f,0.9f,"HiddenNode_Two",1);
    Neuron hiddenNode_Three = new Neuron(0.5f,0.9f,"HiddenNode_Three",1);
    Neuron outputNode = new Neuron(0,0.9f,"OutputNode",3);
    inputNode.connect(hiddenNode_One);
    inputNode.connect(hiddenNode_Two);
    inputNode.connect(hiddenNode_Three);

    hiddenNode_One.connect(outputNode);
    hiddenNode_Two.connect(outputNode);
    hiddenNode_Three.connect(outputNode);

    inputNode.input(1);
}
}
```

Debug Shell | Problems | Terminal | Console ×

<terminated> Neuron [Java Application] C:\Program Files\Java\jdk-18.0.1.1\bin\javaw.exe (05-Dec-2022, 6:29:15 pm – 6:29:17 pm) [pid: 17124]

```
HiddenNode_One:0.9
HiddenNode_Two:0.9
OutputNode:2.4299998
HiddenNode_Three:0.9
InputNode:1.0
```

727821TUAD048
SNEHA.R.S

```java
import java.util.ArrayList;
public class Neuron {
    private float activationValue, weight;
    private String neuronName;
    private ArrayList<Neuron> outputs = new ArrayList<Neuron>();
    private float[] inputs;
    int inputCounter = 0, nInputs;

    public Neuron(float activationValue, float weight, String neuronName, int nInputs){
        this.activationValue = activationValue;
        this.weight = weight;
        this.neuronName = neuronName;
        this.nInputs = nInputs;
        inputs = new float[nInputs];
    }
    public void connect(Neuron neuron){
        outputs.add(neuron);
    }

    public void input(float inputValue){
        inputs[inputCounter] = inputValue;
        inputCounter++;
        if(inputCounter == nInputs){
            fire();
        }
    }

    public void fire(){
        float sum = 0;
        for(int i = 0; i < nInputs; i++){
            sum+=inputs[i];
        }

        float signal = sum*weight;


        if(signal > activationValue){
            for(int i = 0; i < outputs.size(); i++){
                outputs.get(i).input(signal);
            }
        } else{
            for(int i = 0; i < outputs.size(); i++){
                outputs.get(i).input(0);
            }
        }
        System.out.println(neuronName + ":" + signal);
    }
public static void main(String[] args) {
    Neuron inputNode = new Neuron(0,1,"InputNode",1);
    Neuron hiddenNode_One = new Neuron(0.5f,0.9f,"HiddenNode_One",1);
    Neuron hiddenNode_Two = new Neuron(0.5f,0.9f,"HiddenNode_Two",1);
    Neuron hiddenNode_Three = new Neuron(0.5f,0.9f,"HiddenNode_Three",1);
    Neuron outputNode = new Neuron(0,0.9f,"OutputNode",3);
    inputNode.connect(hiddenNode_One);
    inputNode.connect(hiddenNode_Two);
    inputNode.connect(hiddenNode_Three);

    hiddenNode_One.connect(outputNode);
    hiddenNode_Two.connect(outputNode);
    hiddenNode_Three.connect(outputNode);

    inputNode.input(1);
}
}
```

Debug Shell | Problems | Terminal | Console ×

<terminated> Neuron [Java Application] C:\Program Files\Java\jdk-18.0.1.1\bin\javaw.exe (05-Dec-2022, 6:29:15 pm – 6:29:17 pm) [pid: 17124]

```
HiddenNode_One:0.9
HiddenNode_Two:0.9
OutputNode:2.4299998
HiddenNode_Three:0.9
InputNode:1.0
```