```
from pomegranate import *
import numpy as np

Rain = DiscreteDistribution( {"Yes": 0.2, "No": 0.8} )

Sprinkler = ConditionalProbabilityTable(
    [
        ["No", "Off", 0.6],
        ["No", "On", 0.4],
        ["Yes", "Off", 0.99],
        ["Yes", "On", 0.01],
    ], [Rain]
)

Grass = ConditionalProbabilityTable(
    [
        ["Off", "No", "Dry", 1.0],
        ["Off", "No", "Wet", 0.0],
        ["Off", "Yes", "Dry", 0.2],
        ["Off", "Yes", "Wet", 0.8],
        ["On", "No", "Dry", 0.1],
        ["On", "No", "Wet", 0.9],
        ["On", "Yes", "Dry", 0.01],
        ["On", "Yes", "Wet", 0.99],
    ], [Sprinkler, Rain]
)

sRain = Node(Rain, name="Rain")
sSprinkler = Node(Sprinkler, name="Sprinkler")
sGrass = Node(Grass, name="Grass")
```

```
model = BayesianNetwork("Wet Grass Network")
model.add_nodes(sRain, sSprinkler, sGrass)
model.add_edge(sRain, sSprinkler)
model.add_edge(sRain, sGrass)
model.add_edge(sSprinkler, sGrass)
model.bake()

arr = np.array(["Yes", "Off", "Wet"], ndmin=2)
ans = np.e ** model.log_probability(arr)
print("The value of P is =  ",ans)
```