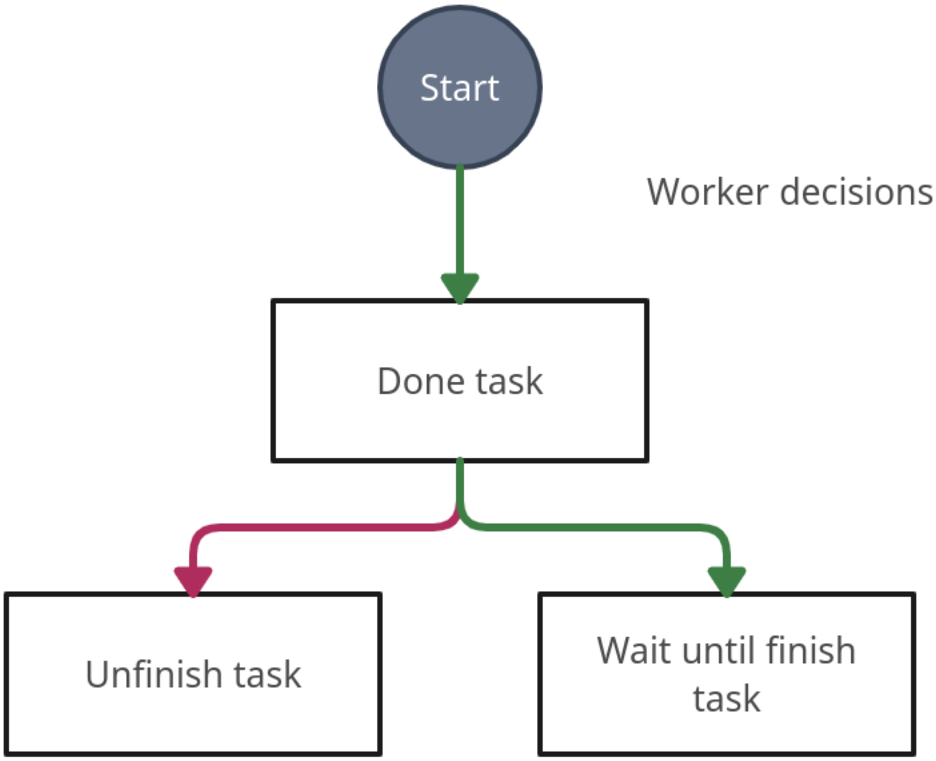
Worker Dicisions

In this notebook, we will see how the good behavior of the worker will affect the increase of his inventory. we have considered the initial balance as 20 ether units. you can see the decision tree of the worker below:

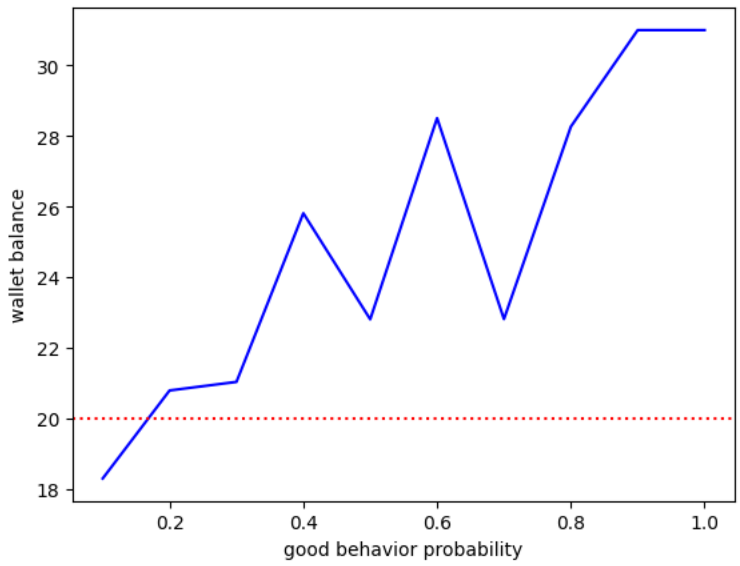


Good behavior is marked with green lines and bad behavior with red lines. variable *decisionProbability* shows the possibility of good behavior from the worker

```
import random

taskProperties = [
    (4, 2, 2),
    (5, 2, 2),
    (7, 3, 3),
    (9, 5, 4),
    (11, 10, 9),
    (15, 5, 6),
    (17, 11, 10),
    (20, 12, 13)
]

workerInitialBalance = 20

def calculateWorkerBalance(decisionProbability):
  if decisionProbability < 0 or decisionProbability > 1:
    raise ValueError(decisionProbability, "Args must be between 0 and 1")
  else:
    result = 0
    for taskProperty in taskProperties:
      rand = random.random()
      salary, requetserProofOfTrust, workerProofOfTrust = taskProperty
      if rand <= decisionProbability:
        result = result + salary
      else:
        repaymentRate = (salary + (requetserProofOfTrust / 4)) / (salary + requetserProofOfTrust)
        result = result - (workerProofOfTrust * (1 - repaymentRate))
    return workerInitialBalance + (result / len(taskProperties))
```

Now in the code below, you can see that for the increase of good behavior from the worker, the balance of his wallet increases. Please note that we have considered the initial balance as 20 units.

```
import matplotlib.pyplot as plt
import numpy as np

probabilities = [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1]
outputs = [calculateWorkerBalance(i) for i in probabilities]

plt.plot(probabilities, outputs, 'b-')
plt.axhline(y=20, color='r', linestyle=':')
plt.xlabel("good behavior probability")
plt.ylabel("wallet balance")
plt.show()
```



Colab paid products - Cancel contracts here

✓ 0s completed at 2:04 PM