Planning, Learning and Decision Making

Homework 1. Markov chains

Group 27

78375 - João Pirralha

84758 - Rafael Ribeiro

1.a)

State space: $\mathcal{X} = \{Station\ A,\ Stop\ 1,\ Stop\ 2,\ Stop\ 3,\ Stop\ 4,\ Stop\ 5,\ Stop\ 6,\ Station\ B\}$

The transition probability matrix P is done in Python as it will be used in the next part of the exercise:

In [1]:
```python
import numpy as np
X = ("Station A", "Stop 1", "Stop 2", "Stop 3", "Stop 4", "Stop 5",
     "Stop 6", "Station B")
P = np.zeros((8, 8))
P[0, 1] = 0.5
P[0, 4] = 0.15
P[0, 5] = 0.35
P[1, 2] = 1
P[2, 3] = 1
P[3, 7] = 1
P[4, 7] = 1
P[5, 6] = 1
P[6, 7] = 1
P[7, 0] = 1
print("Transition probability matrix:")
print(P)
```

```
Transition probability matrix:
[[0.   0.5  0.   0.   0.15 0.35 0.   0.  ]
 [0.   0.   1.   0.   0.   0.   0.   0.  ]
 [0.   0.   0.   1.   0.   0.   0.   0.  ]
 [0.   0.   0.   0.   0.   0.   0.   1.  ]
 [0.   0.   0.   0.   0.   0.   0.   1.  ]
 [0.   0.   0.   0.   0.   0.   1.   0.  ]
 [0.   0.   0.   0.   0.   0.   0.   1.  ]
 [1.   0.   0.   0.   0.   0.   0.   0.  ]]
```

1.b)

To get the probability for all stops at time step t=3, we can multiply the transition probability matrix P by itself two times (power of 3). Then, to get the probabilities corresponding to when the train departed form Station A at t=0, we extract the first line of the resulting matrix.

This is done in Python:

In [2]:
```
prob_3 = np.linalg.matrix_power(P, 3)[0]
print("Probability of the train being in each stop at time step t = 3:")
for stop, prob in zip(X, prob_3):
    print(stop + ": " + str("%g" % prob))
```

```
Probability of the train being in each stop at time step t = 3:
Station A: 0.15
Stop 1: 0
Stop 2: 0
Stop 3: 0.5
Stop 4: 0
Stop 5: 0
Stop 6: 0
Station B: 0.35
```

1.c)

Let τ be the expected waiting time for a passenger in Stop 4 **when the train departs from Station A**. The expected waiting time for a passenger in Stop 4 **when the train has just departed from Stop 4** is then τ plus the time it takes the train to go from Stop 4 to Station A, which is $(10+2)\times 2 = 24$ minutes.

When the train departs from Station A it might arrive at Stop 4:

- in 10 minutes with 0.15 probability (when the train goes directly to Stop 4);
- in $(10+2)\times 5+\tau$ minutes with 0.5 probability (when the train goes through the longest alternative path);
- in $(10+2)\times 4+\tau$ minutes 0.35 probability (when the train goes through the shortest alternative path).

Thus:

$$\tau = 0.15\times 10 + 0.5\times(12\times 5+\tau) + 0.35\times(12\times 4+\tau)$$
$$\Leftrightarrow \tau = 1.5 + 30 + 0.5\tau + 16.8 + 0.35\tau$$
$$\Leftrightarrow \tau = \frac{1.5+30+16.8}{1-0.5-0.35} = 322\, minutes$$

Adding the 24 minutes necessary to get from Stop 4 to Station A:

$$Expected\ waiting\ time = 322 + 24 = 346\, minutes$$

Next we experimentally validade our answer:

In [3]:
```
# Simulation for validation
ITERATIONS = 10 ** 5
STOPS = [n for n in range(8)]
total_time_waiting = 0

for i in range(ITERATIONS):
    stop = 4
    total_time_waiting += 12
    stop = np.random.choice(STOPS, p = P[stop])
    while stop != 4:
        total_time_waiting += 12
        stop = np.random.choice(STOPS, p = P[stop])
    # The passenger boards the train as soon as it arrives
    total_time_waiting -= 2


print("(Simulation) expected waiting time: "
      + "%g" % (total_time_waiting / ITERATIONS)
      + " minutes")
```

```
(Simulation) expected waiting time: 346.853 minutes
```