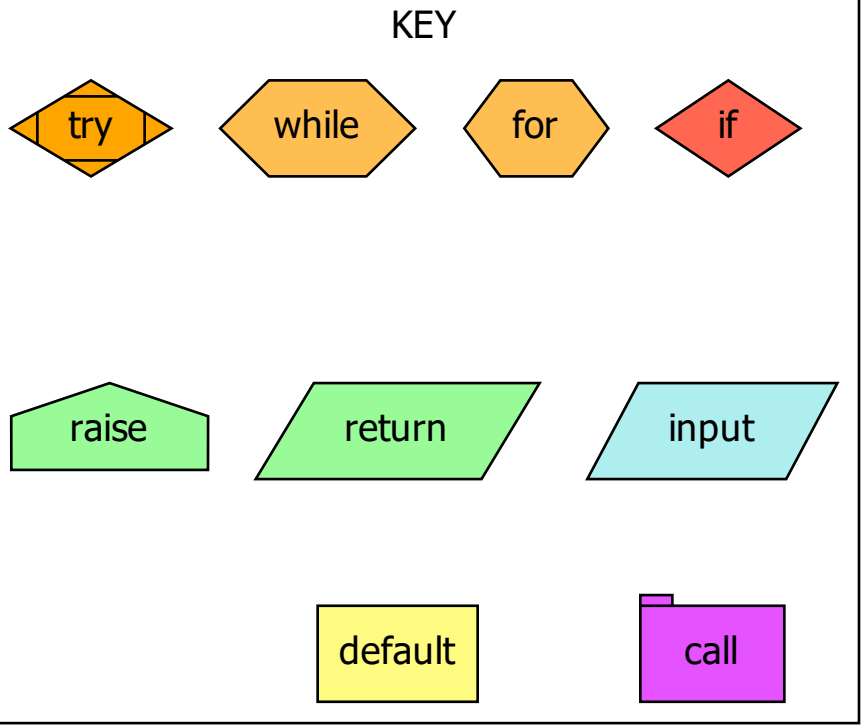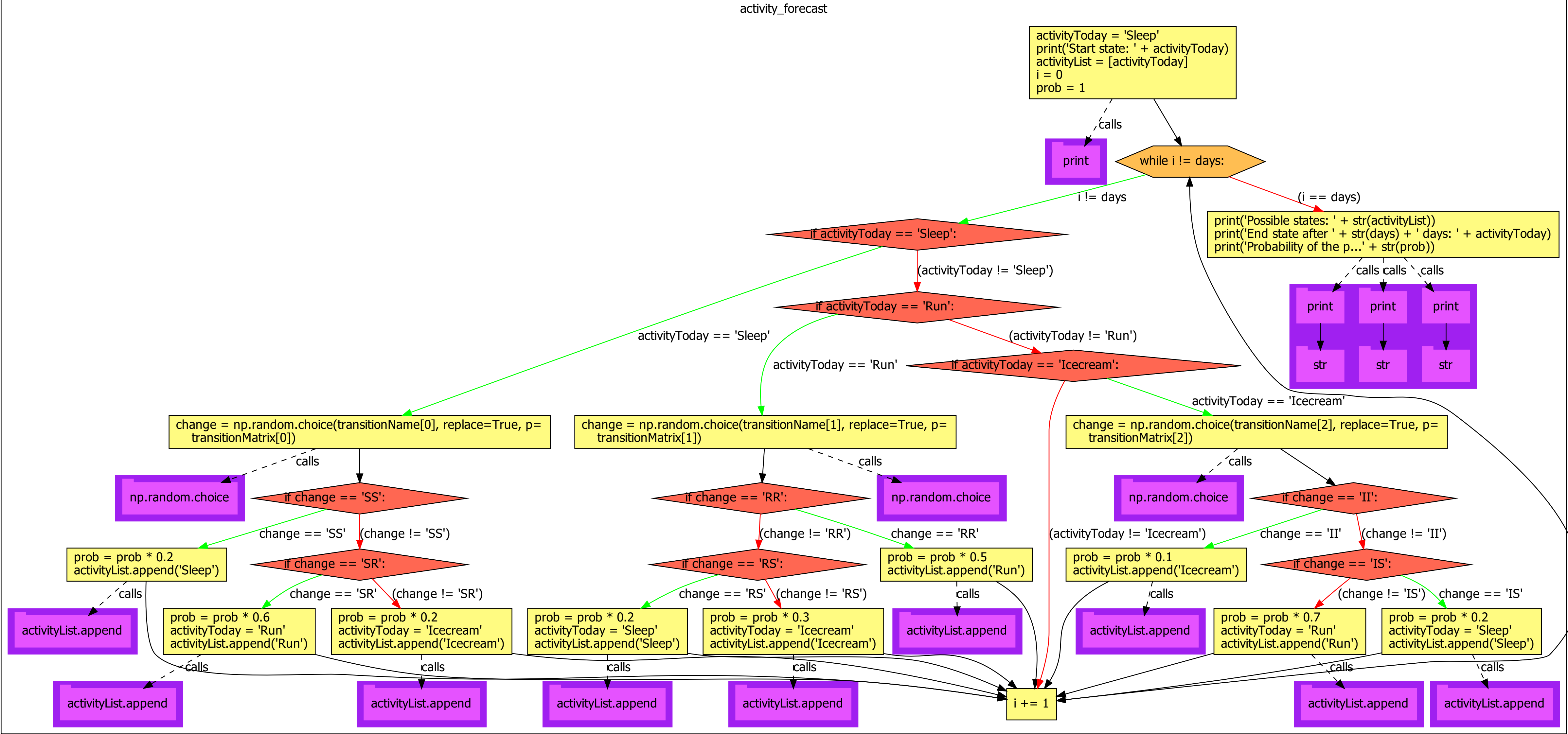```
import numpy as np
import random as rm
states = ['Sleep', 'Icecream', 'Run']
transitionName = [['SS', 'SR', 'SI'], ['RS', 'RR', 'RI'], ['IS', 'IR', 'II']]
transitionMatrix = [[0.2, 0.6, 0.2], [0.1, 0.6, 0.3], [0.2, 0.7, 0.1]]
def activity_forecast(days):...
activity_forecast(2)
```

**activity_forecast** — calls → activity_forecast

**KEY**

- try
- while
- for
- if
- raise
- return
- input
- default
- call

**activity_forecast**

```
activityToday = 'Sleep'
print('Start state: ' + activityToday)
activityList = [activityToday]
i = 0
prob = 1
```

calls → print

while i != days

(i == days) →
```
print('Possible states: ' + str(activityList))
print('End state after ' + str(days) + ' days: ' + activityToday)
print('Probability of the p...' + str(prob))
```
calls → print, str / calls → print, str / calls → print, str

**example**

i != days

if activityToday == 'Sleep':

(activityToday != 'Sleep') →
if activityToday == 'Run':

(activityToday != 'Run') →
if activityToday == 'Icecream':

activityToday == 'Sleep' →
```
change = np.random.choice(transitionName[0], replace=True, p=
transitionMatrix[0])
```
calls → np.random.choice

activityToday == 'Run' →
```
change = np.random.choice(transitionName[1], replace=True, p=
transitionMatrix[1])
```
calls → np.random.choice

activityToday == 'Icecream' →
```
change = np.random.choice(transitionName[2], replace=True, p=
transitionMatrix[2])
```
calls → np.random.choice

if change == 'SS':

change == 'SS' →
```
prob = prob * 0.2
activityList.append('Sleep')
```
calls → activityList.append

(change != 'SS') →
if change == 'SR':

change == 'SR' →
```
prob = prob * 0.6
activityToday = 'Run'
activityList.append('Run')
```

(change != 'SR') →
```
prob = prob * 0.2
activityToday = 'Icecream'
activityList.append('Icecream')
```

if change == 'RR':

change == 'RR' →
```
prob = prob * 0.5
activityList.append('Run')
```
calls → activityList.append

(change != 'RR') →
if change == 'RS':

change == 'RS' →
```
prob = prob * 0.2
activityToday = 'Sleep'
activityList.append('Sleep')
```

(change != 'RS') →
```
prob = prob * 0.3
activityToday = 'Icecream'
activityList.append('Icecream')
```

if change == 'II':

change == 'II' →
```
prob = prob * 0.1
activityList.append('Icecream')
```
calls → activityList.append

(change != 'II') →
if change == 'IS':

(change != 'IS') →
```
prob = prob * 0.7
activityToday = 'Run'
activityList.append('Run')
```
calls → activityList.append

change == 'IS' →
```
prob = prob * 0.2
activityToday = 'Sleep'
activityList.append('Sleep')
```
calls → activityList.append

(activityToday != 'Icecream')

calls → activityList.append (multiple)

i += 1