例 6.2 Python 程序代码

```
import numpy as np
import random as rm


def activity_forecast(days):
    # 选择初始状态
    activityToday = "Sleep"
    activityList = [activityToday]
    i = 0
    prob = 1
    while i != days:
        if activityToday == "Sleep":
            change    =    np.random.choice(transitionName[0],    replace=True,
p=transitionMatrix[0])
            if change == "SS":
                prob = prob * 0.2
                activityList.append("Sleep")
                pass
            elif change == "SR":
                prob = prob * 0.6
                activityToday = "Run"
                activityList.append("Run")
            else:
                prob = prob * 0.2
                activityToday = "Icecream"
                activityList.append("Icecream")
        elif activityToday == "Run":
            change    =    np.random.choice(transitionName[1],    replace=True,
p=transitionMatrix[1])
            if change == "RR":
                prob = prob * 0.5
                activityList.append("Run")
                pass
            elif change == "RS":
                prob = prob * 0.2
                activityToday = "Sleep"
                activityList.append("Sleep")
            else:
                prob = prob * 0.3
                activityToday = "Icecream"
                activityList.append("Icecream")
        elif activityToday == "Icecream":
            change    =    np.random.choice(transitionName[2],    replace=True,
```

```
p=transitionMatrix[2])
            if change == "II":
                prob = prob * 0.1
                activityList.append("Icecream")
                pass
            elif change == "IS":
                prob = prob * 0.2
                activityToday = "Sleep"
                activityList.append("Sleep")
            else:
                prob = prob * 0.7
                activityToday = "Run"
                activityList.append("Run")
        i += 1
    return activity


# 记录每次的 activityList
list_activity = []
count = 0

# `range` 从第一个参数开始数起，一直到第二个参数（不包含）
for iterations in range(1, 10000):
    list_activity.append(activity_forecast(2))

# 查看记录到的所有 `activityList`
# print(list_activity)

# 遍历列表，得到所有最终状态是跑步的 activityList
for smaller_list in list_activity:
    if (smaller_list[2] == "Run"):
        count += 1

# 计算从睡觉状态开始到跑步状态结束的概率
percentage = (count / 10000) * 100
print("The probability of starting at state:'Sleep' and ending at state:'Run'= " +
str(percentage) + "%")
```