

姓名：李宇豪

学号：21305412

周数：2

成绩：

### 1. 程序：

```
import numpy as np

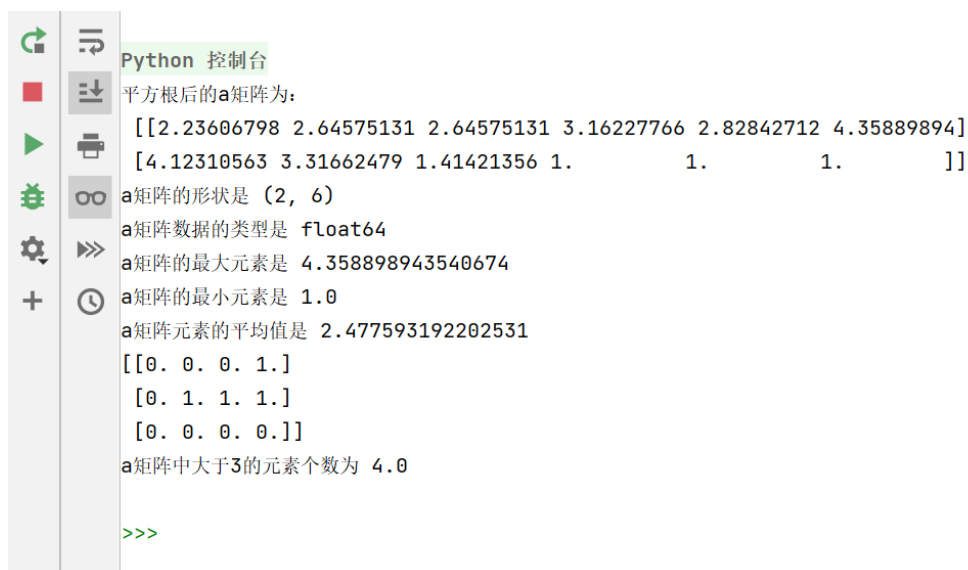
a = np.random.randint(0, 21, (2, 6))
a = np.sqrt(a)
print("平方根后的 a 矩阵为：\n", a)

print("a 矩阵的形状是", a.shape)
print("a 矩阵数据的类型是", a.dtype)
print("a 矩阵的最大元素是", a.max())
print("a 矩阵的最小元素是", a.min())
print("a 矩阵元素的平均值是", a.mean())

a = a.reshape(3, 4)
b = a - 3
b[b > 0] = 1
b[b != 1] = 0

print(b)
print("a 矩阵中大于 3 的元素个数为", b.sum())
```

输出：

A screenshot of a Python Jupyter Notebook console. The left sidebar shows standard Jupyter icons: a green arrow for running, a red square for stopping, a green play button for stepping through, a magnifying glass for search, a gear for settings, and a plus sign for additional actions. The main area displays the output of the code. It starts with the text '平方根后的a矩阵为:' followed by a 2x6 matrix of floating-point numbers. Then, it shows the shape of 'a' as (2, 6), its dtype as float64, its maximum value as 4.358898943540674, its minimum value as 1.0, and its mean as 2.477593192202531. Next, it displays the 3x4 matrix 'b' with values 0.0, 0.0, 0.0, 1.0 in the first row, 0.0, 1.0, 1.0, 1.0 in the second row, and 0.0, 0.0, 0.0, 0.0 in the third row. Finally, it shows that the number of elements in 'a' greater than 3 is 4.0. The prompt '>>>' is visible at the bottom.

```
Python 控制台
平方根后的a矩阵为:
[[2.23606798 2.64575131 2.64575131 3.16227766 2.82842712 4.35889894]
 [4.12310563 3.31662479 1.41421356 1.          1.          1.          ]]

a矩阵的形状是 (2, 6)
a矩阵数据的类型是 float64
a矩阵的最大元素是 4.358898943540674
a矩阵的最小元素是 1.0
a矩阵元素的平均值是 2.477593192202531
[[0. 0. 0. 1.]
 [0. 1. 1. 1.]
 [0. 0. 0. 0.]]
a矩阵中大于3的元素个数为 4.0

>>>
```

## 2. 程序:

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

x = np.linspace(0.01, 5, 1000)
y1=np.log(x)
y2=np.log(5*x)

p1,=plt.plot(x,y1,color="purple",linewidth=2.0,linestyle="-")
p2,=plt.plot(x,y2,color="green",linewidth=2.0,linestyle="-")

plt.xticks(np.arange(0, 5))
plt.yticks([-4,-2,0,2,4])

ax=plt.gca()
ax.spines['right'].set_color('none')
ax.spines['top'].set_color('none')

ax.xaxis.set_ticks_position('bottom')
ax.spines['bottom'].set_position(('data',0))
ax.yaxis.set_ticks_position('left')
ax.spines['left'].set_position(('data', -0.2))

legend=plt.legend([p1,p2],["log(x)", "log(5x)"], fontsize=12, loc
='upper left')

plt.plot([0,3,3],[0,np.log(3),np.log(15)],linewidth=1.5,linest
yle="--")
plt.scatter([3,3],[np.log(3),np.log(15)],50)

plt.show()
```

## 输出:

