

Python 程序设计 实验 8: Numpy 和标准库

注意事项:

- (1) 实验报告提交**截止日期: 2021.05.27, 23:59pm**, 迟交扣 20%, 缺交 0 分。
- (2) 实验报告内容包括: 解决问题的思路与方法(如代码的解释)、遇到的问题以及收获(简单描述即可)、代码运行结果的展示。
- (3) 实验报告提交方法: **blackboard**。
- (4) 提交要求: 实验报告+源代码, 打包上传, 命名: 学号_姓名_实验报告_8。
- (5) **禁止抄袭**, 一经发现 **0 分处理**(包括抄袭者和提供代码或实验报告者)!

1. Numpy 基础: 运行以下代码, 理解每句代码的意思或输出结果(print 语句):

(a)

```
mysqrt = [math.sqrt(x) for x in range(0,5)]
mycrt = [x**(1/3) for x in range(0,5)]

npData = np.array(mysqrt)
print("The shape:", npData.shape)
print("The dimensionality:", npData.ndim)
print("The type:", npData.dtype)

twoDarray = np.array([mysqrt, mycrt])
print("The shape:", twoDarray.shape)
print("The dimensionality:", twoDarray.ndim)
print("The type:", twoDarray.dtype)
```

(b)

```
zeros = np.zeros(3)
zMat = np.zeros((4,3))
ones = np.ones(3)
oMat = np.ones((3,2))
diag = np.eye(4)
rng = np.arange(5)
dm = np.diag(rng)
print(dm.shape)
zMat_re = zMat.reshape(6,2)
```

(c)

```
A = np.random.randint(0,10, size = (3,2))
B = np.random.randint(0,10, size = (3,3,3))
C = np.random.randint(0,10, size = (3,1))
print(A**2)
print(np.sqrt(A))
print(A + C)
print(B + C)
B[:, 0:2 , 0:2 ] -= 20
print(B)
```

(d)

```
from numpy import linalg
A = np.array([[2, 1, -2],[1,-1,-1], [1, 1 ,3]])
b = np.array([3, 0, 12])
x = linalg.solve(A,b)
print(x)
```

2. **Numpy 应用 1:** 给定一个矩阵 $2n \times 2n$ ，将该矩阵分成四个象限（参见示例），然后返回一个新的 2×2 矩阵，包含每个象限的平均值。

例子：

原矩阵：	1, 2, 5, 7	1, 2 5, 7
	4, 1, 8, 0	4, 1 8, 0
	2, 0, 5, 1	-----
	0, 2, 1, 1	2, 0 5, 1
		0, 2 1, 1

新矩阵：

(1+2+4+1)/ 4		(5+7+8+0)/4	=>	2.0	, 5.0
-----				1.0	, 2.0
(2+0+0+2)/4		(5+1+1+1)/4			

3. **Numpy 应用 2:** 仿照课件中利用 numpy 处理图片的方法，选择一张自己喜欢的图片进行处理，方法不限（如调整亮度、彩色变黑白、模糊化等等；鼓励自学图片处理方法，提出自己的方法）。

4. Python 标准库 itertools: 编写函数 `sum0(lst)`, 接受一个包含 8 个整数的列表 `lst`。如果列表中的某些非空子集的总和返回 0, 则返回 `True`。例如, `lst=[-3, 11, 21, 5, 10, 11, 2, 1]` 返回 `True`, 因为非空子集 `[-3, 2, 1]` 中数字加起来总和为 0; 又如 `lst=[2, 3, 4, 5, 6, 7, 8, 9]` 时, 函数返回 `False`。

5. Python 标准库 datetime: 编写函数 `calculate_age`, 计算自出生以来到目前为止, 生活的总天数, 总月数, 总年数, 返回元组 (`days`, `months`, `years`);

6. Python 标准库 collections, sys, os: 统计目前写过的 python 代码。

(1) 把实验、作业、小测的代码文件整理好, 分开放在“作业”文件夹、“实验”文件夹、“实验”文件夹, 三个文件夹放到同一个文件夹“代码”中。

(2) 编写函数, 统计“代码”文件夹中 python 文件的个数 (`file_num`), 写过的代码行数 (`code_line`), 代码中空行的行数 (`space_lines`), 注释的行数 (`comments_lines`, 只统计以 `#` 开头的), 返回元组记录上述信息。

(3) 在 (2) 基础上, 允许用户输入指定的文件或文件夹, 统计输入文件或文件夹的信息。例如, 假设 python 文件名为 `code_stat.py`, 运行方法如下:

```
>>> python code_stat.py 代码/作业          #统计“作业”文件夹
```

```
>>> python code_stat.py 代码/实验/实验 1.py #只统计实验 1.py
```