

现代程序设计第七次作业

1. 库的引入

```
import re
from memory_profiler import profile
from line_profiler import LineProfiler
import pandas as pd
import time
from tqdm import tqdm
import pathlib
#from playsound import playsound
import subprocess
```

2. 要求1 实现一个类，在其中提供一些方法模拟耗时耗内存的一些操作，以测试如下的装饰器（用类或函数实现），如大的数据结构生成、遍历、写入文件序列化等。

创建一个计算素数的类，包含判断是否为素数，测试一部分数中的素数个数，对素数进行保存等方法

```
class Prime:
    def __init__(self,maxnum):
        self.maxnum = maxnum

    def is_prime(self,num):
        if num < 2:
            return False
        elif num == 2:
            return True
        else:
            for i in range(2,num):
                if num%i == 0:
                    return False
            return True

    def prime_test(self):
        count = 0
        for i in range(2,self.maxnum):
            if self.is_prime(i):
                #print(i)
                count += 1
        return count

    def prime_save(self):
        self.prime_list = []
        for i in range(2,self.maxnum):
            if self.is_prime(i):
```

```
self.prime_list.append(i)
```

3. 要求2如果需要知道程序的运行时间、运行进度、内存占用情况，请利用line_profiler、memory_profiler、tqdm等装饰器实现相关功能，要求在程序执行结束后，打印程序的内存占用和运行时间。

利用装饰器的闭包特性，对装饰其进行设计，实现对代码的内存和运行事件监测并返回

```
def process_display(func):
    def wrapper(*args):
        num_list = list(range(*args))
        #print(num_list)
        bar = tqdm(num_list)
        for num in bar:
            bar.set_description("Now get "+str(num))
            # time.sleep(0.0001)
        func(*args)
    return wrapper

def diplay_time(func):
    def wrapper(*args):
        func(*args)
    return wrapper

@profile
def mem_space(maxnum):
    Prime_space = Prime(maxnum)
    Prime_space.prime_save()
    del Prime_space.prime_list

@process_display
@diplay_time
def test_prime(maxnum):
    Prime_test = Prime(maxnum)
    lp = LineProfiler()
    lp_wrapper = lp(Prime.prime_test)
    num_prime = lp_wrapper(Prime_test)
    print("-----")
    print(f"There is {num_prime} prime in it")
    print("-----")
    lp.print_stats()
```

4. 要求三 在程序处理结束后，通常需要将模型或者数据处理结果保存下来。但是，有时会因为路径设置错误（忘记新建文件夹）等原因导致文件无法存储，浪费大量的时间重复运行程序。一种解决方法是在执行程序前

对参数中的路径进行检查。要求利用装饰器函数实现这一功能，接收函数的路径参数，检查路径对应文件夹是否存在，若不存在，则给出提示，并在提示后由系统自动创建对应的文件夹。

利用装饰器对函数在运行之前进行检查，并在检查后对文件夹进行创建

```
def check_path(func):
    def wrapper(*args):
        path = pathlib.Path(*args)
        if path.exists():
            func(*args)
        else:
            print("The filepath which you want to save to doesn't exist.")
            path.mkdir()
    return wrapper

@check_path
def save(path):
    zeros = pd.DataFrame([[0] for i in range(100)])
    zeros.to_excel(path + '/zero.xlsx')
```

5. 要求四在程序运行结束后，可以给用户发送一个通知，比如播放一段音乐等。要求实现对应的装饰器类，在被装饰的函数执行结束后，可以主动播放声音（了解并使用一下playsound或其他声音文件处理的库）。

```
def complete_remind(func):
    def wrapper():
        func()

        subprocess.call(['afplay', r'/Users/xihe/Documents/python_mp2021_project/week7_project/clock.mp3'])
    return wrapper

@complete_remind
def text_remind():
    print("the code has been run completely.")
```

代码测试

由于使用了装饰器所以实际执行的函数就比较简单

```
def main():

    mem_space(100000)

    test_prime(10000)

    save('/Users/xihe/Documents/python_mp2021_project/week7_project/zero')

    text_remind()

main()
```

结果展示

内存占用展示

Line #	Mem usage	Increment	Occurrences	Line Contents
60	89.8 MiB	89.8 MiB	1	@profile
61				def mem_space(maxnum):
62	89.8 MiB	0.0 MiB	1	Prime_space = Prime(maxnum)
63	92.7 MiB	2.9 MiB	1	Prime_space.prime_save()
64	92.7 MiB	0.0 MiB	1	del Prime_space.prime_list

时间占用展示及进度条显示

Now get 9999: 100% |

```
10000/10000 [00:00<00:00, 12555.78it/s]
```

There is 5000 prime in it

Timer unit: 1e-06 s

```
Total time: 0.02108 s
```

File: /Users/xihe/Documents/python mp2021 project/week7 project/decorate class.py

```
Function: prime test at line 29
```

Line #	Hits	Time	Per Hit	% Time	Line Contents
29					def prime_test(self):
30	1	8.0	8.0	0.0	count = 0
31	9999	2783.0	0.3	13.2	for i in range(2,self.maxnum):
32	9998	16765.0	1.7	79.5	if self.is_prime(i):
33					#print(i)
34	5000	1524.0	0.3	7.2	count += 1
35	1	0.0	0.0	0.0	return count

文件保存时若文件夹不存在

The filepath which you want to save to doesn't exist.
并创建了文件夹

文件保存时文件夹存在时

直接将文件写入不报错

代码执行结束后提醒

the code has been run completely.
播放clock.mp3