

目录

介绍	2
必做题	3
1. 寻找水仙花数（模块：numEx，所在文件名 num_hw.py，Level：★）	3
2. 寻找完美数（模块：numEx，所在文件名 num_hw.py，Level：★）	3
3. 百钱百鸡问题（模块：numEx，所在文件名 num_hw.py，Level：★）	3
4. 最大公约数和最小公倍数（模块：numEx，所在文件名 num_hw.py，Level：★）	4
5. 回文数（模块：numEx，所在文件名 num_hw.py，Level：★）	4
6. 素数（模块：numEx，所在文件名 num_hw.py，Level：★）	4
7. 约瑟夫环问题（模块：numEx，所在文件名 num_hw.py，Level：★★）	4
8. 万年历（模块：numEx，所在文件名 num_hw.py，Level：★★）	5
9. 两地之间距离计算（模块：numEx，所在文件名 num_hw.py，Level：★）	5
10. 计算 Fibonacci 序列的值（模块：numEx，所在文件名 num_hw.py，Level：★）	6
代码建议	7

必做题

1. 寻找水仙花数（模块：numEx，所在文件名 num_hw.py，Level: ★）

水仙花数（Narcissistic number）是指一个 3 位数，它的每个位上的数字的 3 次幂之和等于它本身（例如： $1^3 + 5^3 + 3^3 = 153$ ）。本题要求寻找所有的水仙花数。

函数原型：def narcissistic_number()

返回值：返回一个 list，包含了所寻找到的全部水仙花数的数值，要求这些数从小到大排列。每一个数都应当为整形，如[153, 370, 371]。

2. 寻找完美数（模块：numEx，所在文件名 num_hw.py，Level: ★）

完全数（Perfect number），又称完美数或完备数，是一些特殊的自然数。它所有的真因子（即除了自身以外的约数）的和（即因子函数），恰好等于它本身。如果一个数恰好等于它的因子之和，则称该数为“完全数”。第一个完全数是 6，它有约数 1、2、3、6，除去它本身 6 外，其余 3 个数相加， $1+2+3=6$ 。第二个完全数是 28，它有约数 1、2、4、7、14、28，除去它本身 28 外，其余 5 个数相加， $1+2+4+7+14=28$ 。第三个完全数是 496，有约数 1、2、4、8、16、31、62、124、248、496，除去其本身 496 外，其余 9 个数相加， $1+2+4+8+16+31+62+124+248=496$ 。题目要求寻找参数规定所有完美数。

函数原型：def perfect_number(limit=1000)

参数 limit：整数，搜寻的上限，比如 limit=1000，表示寻找 1-1000 之间所有的完美数。1000 为 limit 的默认参数。注意处理异常参数

返回值：返回一个 list，包含了函数所寻找到的全部完美数的数值，要求这些数从小到大排列。每一个数都应当为整形，如[6, 28, 496]。如果参数异常，返回错误“Parameter Error.”

3. 百钱百鸡问题（模块：numEx，所在文件名 num_hw.py，Level: ★）

鸡翁一值钱五，鸡母一值钱三，鸡雏三值钱一。百钱买百鸡，问鸡翁、鸡母、鸡雏各几何？

函数原型：def buy_chicken()

返回值：返回一个 list，数列的元素为三元组，代表（鸡翁、鸡母、鸡雏）的数量，如：[[0,25,75],[4,18,78],[8,11,81]]，表示返回三组解，每一组解以三元 list 表示。

4. 最大公约数和最小公倍数（模块：numEx，所在文件名 num_hw.py，Level: ★）

（1） 函数原型：def gcd(x, y)，求取最大公约数

（2） 函数原型：def lcm(x, y)，求取最小公倍数

参数 x, y：正整数

返回值：正整数，其中 gcd(x, y) 返回 x 与 y 的最大公约数，lcm(x, y) 求取 x 与 y 的最小公倍数。如果参数异常，返回错误“Parameter Error.”

5. 回文数（模块：numEx，所在文件名 num_hw.py，Level: ★）

如果一个数反过来与原数相同，那么这就是一个回文数。比如，121 就是一个回文数，-121 不是一个回文数。设计函数验证一个数是否为回文数。

函数原型：def is_palindrome_number(n)

参数 n：输入待测试的数字，可能是正数、负数、整数、浮点数等数值

返回值：布尔型，如果这个数是回文数返回 True，否则返回 False

6. 素数（模块：numEx，所在文件名 num_hw.py，Level: ★）

素数指的是除了 1 和它本身以外没有其他因数的数。设计函数验证一个数是否为素数。

函数原型：def is_prime_num(n)

参数 n：正整数，输入待测试的数字。负数、小数归为异常参数。

返回值：布尔型，如果这个数是回文数返回 True，否则返回 False。如果参数异常，返回错误“Parameter Error.”

7. 约瑟夫环问题（模块：numEx，所在文件名 num_hw.py，Level: ★★）

有 15 个基督徒和 15 个非基督徒在海上遇险，为了能让一部分人活下来不得不将其中 15 个人扔到海里面去，有个人想了个办法就是大家围成一个圈，由某个人开始从 1 报数，报到 9 的人就扔到海里面，他后面的人接着从 1 开始报数，报到 9 的人继续扔到海里面，直到扔掉 15 个人。由于上帝的保佑，15 个基督徒都幸免于难，问这些人最开始是怎么站的，

哪些位置是基督徒哪些位置是非基督徒。假设初始有 m 人，最后有 n 人存活，设计函数计算存活者的初始站立位置。

函数原型：def jose_prob(n, m)

参数 n ：正整数，小于 m ，表示基督徒的人数，即存活下来的人数。

参数 m ：正整数，表示初始所有的人数。

返回值：返回一个长度为 m 的 list，数值为 0 或 1，其中 1 代表基督徒，0 代表非基督徒。该 list 即指明了基督徒初始站立的位置。如果参数异常，返回错误“Parameter Error.”

8. 万年历（模块：numEx，所在文件名 num_hw.py，Level：★★）

给出一个年月日，计算这一天是那一年的第几天。

函数原型：def calendar(year, month, day)

参数 year：四位正整数，年，如 2000。

参数 month：1-12 的正整数，月

参数 day：1-31 的正整数，日

返回值：正整数，表示这一天是那一年的第几天。如果参数异常，返回错误“Parameter Error.”

9. 两地之间距离计算（模块：numEx，所在文件名 num_hw.py，Level：★）

利用 Python 实现地球上两点之间的距离计算，地球上点的位置以经纬度坐标形式提供。距离计算采用 Haversine 公式：

$$\begin{aligned} d &= 2r \arcsin\left(\sqrt{\text{hav}(\varphi_2 - \varphi_1) + \cos(\varphi_1) \cos(\varphi_2) \text{hav}(\lambda_2 - \lambda_1)}\right) \\ &= 2r \arcsin\left(\sqrt{\sin^2\left(\frac{\varphi_2 - \varphi_1}{2}\right) + \cos(\varphi_1) \cos(\varphi_2) \sin^2\left(\frac{\lambda_2 - \lambda_1}{2}\right)}\right) \end{aligned}$$

这里 r 是地球半径 6371Km， (φ, λ) 代表点的（纬度，经度）坐标。

参考网站：https://en.wikipedia.org/wiki/Haversine_formula

完成距离计算函数：

函数原型：def sphere_distance(p1, p2)

参数 p1：tuple 元组类型，二元组，（纬度，经度），坐标精确到小数点后 7 位

参数 p2: tuple 元组类型，二元组，（纬度，经度），坐标精确到小数点后 7 位

纬度取值范围：[0-90]，经度取值范围：[0-180]，单位均为角度；而 Haversine 公式计算采用的是弧度，注意转换。

返回值：如果输入的坐标数据合规，则返回两点之间的距离，单位为 Km，保留两位小数；如果输入的坐标不合规，返回错误“Parameter Error.”

10. 计算 Fibonacci 序列的值（模块：numEx，所在文件名 num_hw.py，Level: ★）

利用 Python 实现 Fibonacci 序列值的计算。实现两个函数：

（1）递归版本的 Fibonacci 序列值计算

函数原型：def fibonacci_recursion(number)

参数 number: Fibonacci 序列的第 number 项，number 为大于 0 的整数。

返回值：如果参数合规，则返回 Fibonacci 序列的第 number 项的值；如果参数不合规，返回错误“Parameter Error.”。

（2）循环版本的 Fibonacci 序列值计算

函数原型：def fibonacci_loop(number)

参数 number: Fibonacci 序列的第 number 项，number 为大于 0 的整数。

返回值：如果参数合规，则返回 Fibonacci 序列的第 number 项的值；如果参数不合规，返回错误“Parameter Error.”。

问题：

（1）查看 fibonacci_loop(36)与 fibonacci_recursion(36)的运行时间，哪个运行快？

（2）fibonacci_recursion 版本支持的最大输入是多少？最大值如何更改？