

# Assignment #1: 拉齐大家Python水平

---

Updated 2027 GMT+8 Mar 5, 2024

2024 spring, Compiled by ==靳禹安 化学与分子工程学院==

## 说明：

- 1) 数算课程的先修课是计概，由于计概学习中可能使用了不同的编程语言，而数算课程要求Python语言，因此第一周作业练习Python编程。如果有同学坚持使用C/C++，也可以，但是建议也要会Python语言。
  - 2) 请把每个题目解题思路（可选），源码Python, 或者C++（已经在Codeforces/Openjudge上AC），截图（包含Accepted），填写到下面作业模版中（推荐使用 typora <https://typoraio.cn>，或者用word）。AC 或者没有AC，都请标上每个题目大致花费时间。
  - 3) 课程网站是Canvas平台, <https://pku.instructure.com>, 学校通知3月1日导入选课名单后启用。作业写好后，保留在自己手中，待3月1日提交。
- 提交时候先提交pdf文件，再把md或者doc文件上传到右侧“作业评论”。Canvas需要有同学清晰头像、提交文件有pdf、"作业评论"区有上传的md或者doc附件。
- 4) 如果不能在截止前提交作业，请写明原因。

## 编程环境

==（请改为同学的操作系统、编程环境等）==

操作系统：macOS Sonoma 14.0

Python编程环境：Visual Studio Code, Python 3.8.18 (Miniconda)

## 1. 题目

---

### 20742: 泰波拿契數

<http://cs101.openjudge.cn/practice/20742/>

思路：

递归，尝试使用装饰器缓存输出。

代码

```
from functools import lru_cache

@lru_cache
def tribonacci(index: int) -> int:
```

```

if index == 0:
    return 0
if index in (1, 2):
    return 1
return tribonacci(index - 1) + tribonacci(index - 2) + tribonacci(index - 3)

if __name__ == "__main__":
    n = int(input())
    print(tribonacci(n))

```

代码运行截图 == (至少包含有"Accepted") ==

状态: **Accepted**

源代码

```

from functools import lru_cache

@lru_cache
def t_n(n: int) -> int:
    if n == 0:
        return 0
    elif n == 1 or n == 2:
        return 1
    else:
        return t_n(n - 1) + t_n(n - 2) + t_n(n - 3)

if __name__ == "__main__":
    n = int(input())
    print(t_n(n))

```

基本信息

#: 43941223  
 题目: 20742  
 提交人: 22n2200011765  
 内存: 3588kB  
 时间: 21ms  
 语言: Python3  
 提交时间: 2024-02-20 18:21:40

## 58A. Chat room

greedy/strings, 1000, <http://codeforces.com/problemset/problem/58/A>

思路:

逐个字符判断, 成功就继续判断下个字符, 直到遍历完。

代码

```

def hello(word: str) -> str:
    HELLO = "hello"
    pos = 0
    for char in word:
        if pos < 5 and char == HELLO[pos]:
            pos += 1
        if pos == 5:
            return "YES"
    return "NO"

```

```
if __name__ == "__main__":
    inp = input()
    print(hello(inp))
```

代码运行截图 == （至少包含有"Accepted"） ==

General										
#	Author	Problem	Lang	Verdict	Time	Memory	Sent	Judged		
247467791	Practice: Pixdane	<a href="#">58A</a> - 10	PyPy 3-64	Accepted	77 ms	24 KB	2024-02-20 14:33:34	2024-02-20 14:33:34	★	Compare

## 118A. String Task

implementation/strings, 1000, <http://codeforces.com/problemset/problem/118/A>

思路：

借鉴了2022年计概时的思路，代码较简短但可读性较差。

代码

```
word = input()
print('.', '.'.join(filter(lambda x: not x in "aoyeui", word.lower())) , sep='')
```

代码运行截图 == （AC代码截图，至少包含有"Accepted"） ==

General										
#	Author	Problem	Lang	Verdict	Time	Memory	Sent	Judged		
247570666	Practice: Pixdane	<a href="#">118A</a> - 13	PyPy 3-64	Accepted	124 ms	16 KB	2024-02-21 09:06:02	2024-02-21 09:06:02	★	Compare

## 22359: Goldbach Conjecture

<http://cs101.openjudge.cn/practice/22359/>

思路：

并没有卡时间，于是使用了普通的质数判断。

## 代码

```
import math

def is_prime(num: int) -> bool:
    if num == 1:
        return False
    for i in range(2, int(math.sqrt(num) + 1)):
        if num % i == 0:
            return False
    return True

def goldbach(num: int):
    for i in range(2, num//2 + 1):
        j = num - i
        if is_prime(i) and is_prime(j):
            return (i, j)

if __name__ == "__main__":
    n = int(input())
    print(*goldbach(n))
```

代码运行截图 == (AC代码截图, 至少包含有"Accepted") ==

状态: **Accepted**

源代码

```
#22359: Goldbach Conjecture

import math

def is_prime(num: int) -> bool:
    if num == 1:
        return False
    for i in range(2, int(math.sqrt(num) + 1)):
        if num % i == 0:
            return False
    return True

def goldbach(num: int):
    for i in range(2, num//2 + 1):
        j = num - i
        if is_prime(i) and is_prime(j):
            return (i, j)

if __name__ == "__main__":
    n = int(input())
    print(*goldbach(n))
```

基本信息

#: 44081577  
题目: 22359  
提交人: 22n2200011765  
内存: 3596kB  
时间: 22ms  
语言: Python3  
提交时间: 2024-03-05 20:11:43

# 23563: 多项式时间复杂度

<http://cs101.openjudge.cn/practice/23563/>

思路：

对字符串分割处理，得到最高次幂然后输出。

代码

```
def str_parser(word: str):
    words = [tuple(map(lambda x: 1 if x == ' ' else int(x), i.split("n^")) for i in word.split('+'))]
    return [i[1] for i in words if i[0] != 0]

if __name__ == "__main__":
    inp = str_parser(input())
    print(f"n^{max(*inp, 0)}")
```

代码运行截图 == (AC代码截图，至少包含有"Accepted") ==

状态: Accepted

源代码

```
def str_parser(word: str):
    words = [tuple(map(lambda x: 1 if x == ' ' else int(x), i.split("n^")) for i in word.split('+'))]
    return [i[1] for i in words if i[0] != 0]

if __name__ == "__main__":
    inp = str_parser(input())
    print(f"n^{max(*inp, 0)}")
```

基本信息

#: 44081609  
题目: 23563  
提交人: 22n2200011765  
内存: 3632kB  
时间: 22ms  
语言: Python3  
提交时间: 2024-03-05 20:13:22

# 24684: 直播计票

<http://cs101.openjudge.cn/practice/24684/>

思路：

遍历列表求出现最多次的选项。

代码

```
def max_tickets(xs):
    res = []
    maximum = 0
    for x in set(xs):
        count = xs.count(x)
        if count > maximum:
            res = [x]
            maximum = count
        elif count == maximum:
            res.append(x)
    return res

if __name__ == "__main__":
    nums = [int(x) for x in input().split()]
    print(*sorted(max_tickets(nums)))
```

代码运行截图 == (AC代码截图, 至少包含有"Accepted") ==

状态: Accepted

源代码

```
def max_tickets(xs):
    res = []
    maximum = 0
    for x in set(xs):
        count = xs.count(x)
        if count > maximum:
            res = [x]
            maximum = count
        elif count == maximum:
            res.append(x)
    return res

if __name__ == "__main__":
    nums = [int(x) for x in input().split()]
    print(*sorted(max_tickets(nums)))
```

基本信息

#: 44081673  
题目: 24684  
提交人: 22n2200011765  
内存: 14588kB  
时间: 182ms  
语言: Python3  
提交时间: 2024-03-05 20:16:39

## 2. 学习总结和收获

在22计概群里听说老师有开数算后，我马上在选课人数较少时补选了数算。由于化院不必修数算B，时隔一年才再次选闫老师的课程。目前正在尽力拾回python语法和算法。