

Assignment #2: 编程练习

Updated 0953 GMT+8 Feb 24, 2024

2024 spring, Compiled by 李鹏辉, 元培学院

说明:

1) The complete process to learn DSA from scratch can be broken into 4 parts:

- Learn about Time and Space complexities
- Learn the basics of individual Data Structures
- Learn the basics of Algorithms
- Practice Problems on DSA

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

Windows 10 Home, PyCharm 2022.3.2 (Community Edition)

操作系统: macOS Ventura 13.4.1 (c)

Python编程环境: Spyder IDE 5.2.2, PyCharm 2023.1.4 (Professional Edition)

C/C++编程环境: Mac terminal vi (version 9.0.1424), g++/gcc (Apple clang version 14.0.3, clang-1403.0.22.14.1)

1. 题目

27653: Fraction类

<http://cs101.openjudge.cn/practice/27653/>

思路: 约30分钟。

代码

```
1 def reduce(a, b):
2     min_num = min(a, b)
3     for k in range(2, min_num+1):
4         if a % k == 0 and b % k == 0:
5             return reduce(a//k, b//k)
6     return a, b
7
8
9 class Fraction:
10
11     def __init__(self, num, den):
12         self.num = num
13         self.den = den
14
15     def to_add(self, another):
16         num = self.num * another.den + self.den * another.num
17         den = self.den * another.den
18         self.num, self.den = reduce(abs(num), abs(den))
19         if num < 0: self.num = -self.num
20         if den < 0: self.den = -self.den
21
22     def __str__(self):
23         return f'{self.num}/{self.den}'
24
25
26 num_list = input().split()
27 first_fraction = Fraction(int(num_list[0]), int(num_list[1]))
28 second_fraction = Fraction(int(num_list[2]), int(num_list[3]))
29 first_fraction.to_add(second_fraction)
30 print(first_fraction)
```

代码运行截图

状态: Accepted

源代码

```
def reduce(a, b):
    min_num = min(a, b)
    for k in range(2, min_num+1):
        if a % k == 0 and b % k == 0:
            return reduce(a//k, b//k)
    return a, b
```

基本信息

#: 44007691
题目: 27653
提交人: 2100017777_李鹏辉
(2100017777)
内存: 3584kB
时间: 19ms
语言: Python3
提交时间: 2024-02-28 19:16:51

04110: 圣诞老人的礼物-Santa Clau's Gifts

greedy/dp, <http://cs101.openjudge.cn/practice/04110>

思路: 约50分钟。最后打印时一开始未采用格式化输出, 而是使用round函数。

代码

```
1 def q2_parser():
2     sort, max_weight = map(int, input().split())
3     candies = []
4     for _ in range(sort):
5         value, weight = map(int, input().split())
6         candies.append([value/weight, value, weight])
7     candies.sort(key=lambda x: x[0], reverse=True)
8     return max_weight, candies
9
10
11 def value_calculator(max_weight, candies):
12     total_value = 0
13     total_weight = 0
14     for candy in candies:
15         if max_weight > total_weight:
16             if max_weight >= total_weight + candy[2]:
17                 total_value += candy[1]
18                 total_weight += candy[2]
19             else:
20                 total_value += candy[0] * (max_weight - total_weight)
21                 break
22         else: break
23     return total_value
24
25
26 print("{:.1f}".format(value_calculator(*q2_parser())))
```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

源代码

```
def q2_parser():
    sort, max_weight = map(int, input().split())
    candies = []
    for _ in range(sort):
        value, weight = map(int, input().split())
        candies.append([value/weight, value, weight])
    candies.sort(key=lambda x: x[0], reverse=True)
    return max_weight, candies
```

基本信息

#: 44066469
题目: 04110
提交人: 2100017777_李鹏辉
(2100017777)
内存: 3540kB
时间: 20ms
语言: Python3
提交时间: 2024-03-04 16:30:33

18182: 打怪兽

implementation/sortings/data structures, <http://cs101.openjudge.cn/practice/18182/>

思路:

代码

```
1 | #
2 |
```

代码运行截图 (AC代码截图, 至少包含有"Accepted")

230B. T-primes

binary search/implementation/math/number theory, 1300, <http://codeforces.com/problemset/problem/230/B>

思路: 约1小时。最开始没有用埃氏筛法导致超时, 换到第6题看到提示后全部重写。

代码

```
1 | import math
2 |
3 | def sieve(n):
4 |     sieve_list = [False, False] + [True] * (n - 1)
5 |     for i in range(2, int(n**0.5)):
6 |         if sieve_list[i]:
7 |             for j in range(i*i, n+1, i):
8 |                 sieve_list[j] = False
9 |     return sieve_list
10 |
11 |
12 | sieve_list = sieve(1000000)
13 | ignored = input()
14 | nums = map(int, input().split())
15 | for i in nums:
16 |     print('YES' if math.sqrt(i) == int(math.sqrt(i)) and
        sieve_list[int(math.sqrt(i))] else 'NO')
```

代码运行截图 (AC代码截图, 至少包含有"Accepted")

```
import math

def sieve(n):
    sieve_list = [False, False] + [True] * (n - 1)
    for i in range(2, int(n**0.5)):
        if sieve_list[i]:
            for j in range(i*i, n+1, i):
                sieve_list[j] = False
    return sieve_list

sieve_list = sieve(1000000)
ignored = input()
nums = map(int, input().split())
for i in nums:
    print('YES' if math.sqrt(i) == int(math.sqrt(i)) and sieve_list[int(math.sqrt(i))] else 'NO')
```

1364A. XXXXX

brute force/data structures/number theory/two pointers, 1200, <https://codeforces.com/problemset/problem/1364/A>

思路：

代码

```
1 | #
2 |
```

代码运行截图 (AC代码截图, 至少包含有"Accepted")

18176: 2050年成绩计算

<http://cs101.openjudge.cn/practice/18176/>

思路：在第4题的基础上再多花约20分钟。

代码

```
1 | import math
2 |
3 | def sieve(n):
4 |     sieve_list = [False, False] + [True] * (n - 1)
5 |     for i in range(2, int(math.sqrt(n))):
6 |         if sieve_list[i]:
```

```

7         for j in range(i*i, n+1, i):
8             sieve_list[j] = False
9         return sieve_list
10
11
12 def is_T_prime(i):
13     if i < 4:
14         return False
15     if math.sqrt(i) == int(math.sqrt(i)) and sieve_list[int(math.sqrt(i))]:
16         return True
17     return False
18
19
20 def average_calculate(scores):
21     total = 0
22     for score in scores:
23         if is_T_prime(score):
24             total += score
25     if total == 0: return 0
26     return "{:.2f}".format(total / len(scores))
27
28
29 sieve_list = sieve(10000)
30 students_num = int(input().split()[0])
31 for _ in range(students_num):
32     scores = list(map(int, input().split()))
33     print(average_calculate(scores))

```

代码运行截图 (AC代码截图, 至少包含有"Accepted")

状态: Accepted

源代码

```

import math

def sieve(n):
    sieve_list = [False, False] + [True] * (n - 1)
    for i in range(2, int(math.sqrt(n))):
        if sieve_list[i]:
            for j in range(i*i, n+1, i):
                sieve_list[j] = False
    return sieve_list

```

基本信息

#: 44067293
 题目: 18176
 提交人: 2100017777_李鹏辉
 (2100017777)
 内存: 4300kB
 时间: 68ms
 语言: Python3
 提交时间: 2024-03-04 17:49:54

2. 学习总结和收获

如果作业题目简单, 有否额外练习题目, 比如: OJ"2024spring每日选做"、CF、LeetCode、洛谷等网站题目。

第2题在输出细节上花费了大量时间, 之后要更精确地理解题目的输出要求, 如果读不懂就直接看答案的输出方式好了。第4题与第6题的一脉相承再次说明了写函数的重要性!