Assignment #2: 编程练习

Updated 0953 GMT+8 Feb 24, 2024

2024 spring, Complied by 周添 物理学院

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。

1. 题目

27653: Fraction类

http://cs101.openjudge.cn/2024sp_routine/27653/

代码

```
def find_m1(x, y):
    a, b = max(x, y), min(x, y)
    lol = 0
    for i in range(1, b+1):
        if a*i % b == 0:
            lol = i
    return a*lol


def find_m2(x, y):
    a, b = max(x, y), min(x, y)
    lpl = 1
    for i in range(b, 1, -1):
        if a % i == 0 and b % i == 0:
            lpl = i
    return lpl
```

```
a1, a2, b1, b2 = map(int, input().split())
c2 = find_m1(a2, b2)
c1 = int(a1*c2/a2 + b1*c2/b2)
lck = find_m2(c1, c2)
print(f'{int(c1/lck)}/{int(c2/lck)}')
```

状态: Accepted

源代码

```
def find_m1(x, y):
    a, b = max(x, y), min(x, y)
    lol = 0
    for i in range(1, b+1):
        if a*i % b == 0:
            lol = i
    return a*lol


def find_m2(x, y):
    a, b = max(x, y), min(x, y)
    lpl = 1
    for i in range(b, 1, -1):
        if a % i == 0 and b % i == 0:
            lpl = i
    return lpl


a1, a2, b1, b2 = map(int, input().split())
c2 = find_m1(a2, b2)
c1 = int(a1*c2/a2 + b1*c2/b2)
lck = find_m2(c1, c2)
print(f'{int(c1/lck)}/{int(c2/lck)}')
```

基本信息

#:	43953399
题目:	27653
提交人:	23n2300011538
内存:	3648kB
时间:	20ms
语言:	Python3
提交时间:	2024-02-21 20:38:39

04110: 圣诞老人的礼物-Santa Clau's Gifts

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

代码

```
# Compiled by zhou_tian 2300011538 from pku school of physics

def ke(ele):
    return ele[0]


nu,su = input().split()
nu = int(nu)
su = int(su)
c = []
for i in range(nu):
    pri,wei = input().split()
    wei = int(wei)
    pri = int(pri)
    val = pri/wei
```

```
    c.append([val, wei])
c.sort(key=ke, reverse=True)
i = 0
sum_pri = 0
while 1 and i < nu:
    if su > int(c[i][1]):
        su -= int(c[i][1])
        sum_pri += c[i][1]*c[i][0]
        i += 1
    else:
        sum_pri += c[i][0]*su
        break
print("%.1f" % sum_pri)
```

状态: Accepted

源代码

```
# Compiled by zhou_tian 2300011538 from pku school of physics

def ke(ele):
    return ele[0]


nu,su = input().split()
nu = int(nu)
su = int(su)
c = []
for i in range(nu):
    pri,wei = input().split()
    wei = int(wei)
    pri = int(pri)
    val = pri/wei
    c.append([val, wei])
c.sort(key=ke, reverse=True)
i = 0
sum_pri = 0
while 1 and i < nu:
    if su > int(c[i][1]):
        su -= int(c[i][1])
        sum_pri += c[i][1]*c[i][0]
        i += 1
    else:
        sum_pri += c[i][0]*su
        break
print("%.1f" % sum_pri)
```

基本信息

#:	41738817
题目:	04110
提交人:	23n2300011538
内存:	7176kB
时间:	37ms
语言:	Python3
提交时间:	2023-10-18 00:37:12

18182: 打怪兽

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

代码

```
from collections import defaultdict


def beat_the_monster():
    n, limit, hp = map(int, input().split())
    skill_data = defaultdict(list)

    for _ in range(n):
```

```
        num1, num2 = map(int, input().split())
        skill_data[num1].append(num2)

    keys = sorted(skill_data.keys())

    for key in keys:
        skills = skill_data[key]
        skills.sort(reverse=True)
        for i in range(min(limit, len(skills))):
            hp -= skills[i]
            if hp <= 0:
                print(key)
                return
    print('alive')
    return


case_n = int(input())
for i in range(case_n):
    beat_the_monster()
```

状态: Accepted

源代码

```
from collections import defaultdict


def beat_the_monster():
    n, limit, hp = map(int, input().split())
    skill_data = defaultdict(list)

    for _ in range(n):
        num1, num2 = map(int, input().split())
        skill_data[num1].append(num2)

    keys = sorted(skill_data.keys())

    for key in keys:
        skills = skill_data[key]
        skills.sort(reverse=True)
        for i in range(min(limit, len(skills))):
            hp -= skills[i]
            if hp <= 0:
                print(key)
                return
    print('alive')
    return


case_n = int(input())
for i in range(case_n):
    beat_the_monster()
```

基本信息

#:	42192946
题目:	18182
提交人:	23n2300011538
内存:	3780kB
时间:	71ms
语言:	Python3
提交时间:	2023-11-02 22:04:56

230B. T-primes

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

代码

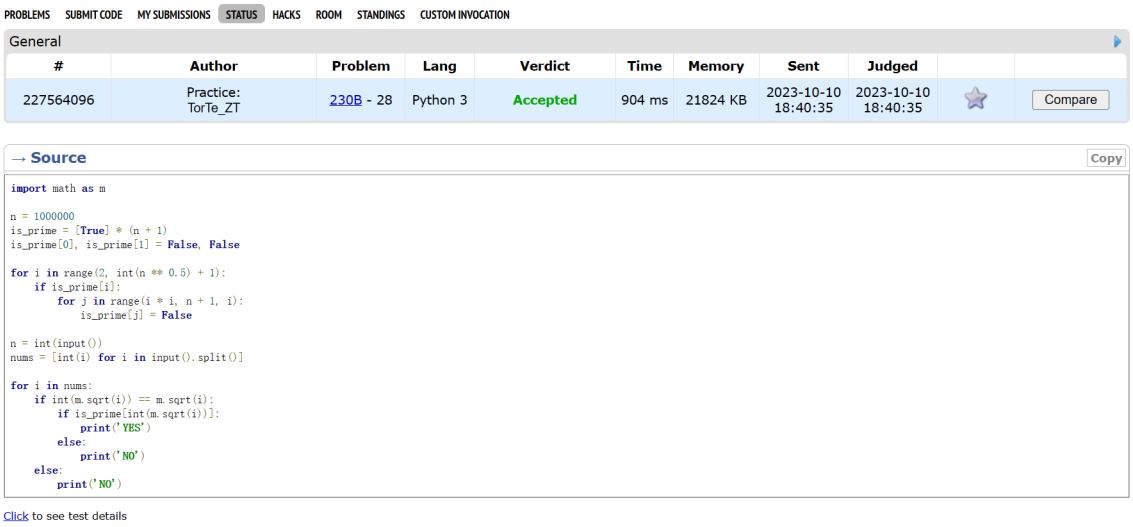
```
import math as m

n = 1000000
is_prime = [True] * (n + 1)
is_prime[0], is_prime[1] = False, False

for i in range(2, int(n ** 0.5) + 1):
    if is_prime[i]:
        for j in range(i * i, n + 1, i):
            is_prime[j] = False

n = int(input())
nums = [int(i) for i in input().split()]

for i in nums:
    if int(m.sqrt(i)) == m.sqrt(i):
        if is_prime[int(m.sqrt(i))]:
            print('YES')
        else:
            print('NO')
    else:
        print('NO')
```



1364A. XXXXX

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

代码

```
#Compiled by zt 2300011538 from pku phy

def is_divisible(nu, number_hated, numbers):
    if sum(numbers) % number_hated != 0:
        return 0
    else:
        for i in range(nu//2+1):
            if numbers[i] % number_hated != 0 or numbers[nu-1-i] % number_hated
!= 0:
                return i+1
        return -1


n = int(input())

for i in range(n):
    nu, number_hated = map(int, input().split())
    numbers = [int(j) for j in input().split()]
    minus = is_divisible(nu, number_hated, numbers)
    print(nu-minus if minus >= 0 else minus)
```

18176: 2050年成绩计算

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

代码

```
prime_list = [True for i in range(10001)]
prime_list[1] = False

for i in range(1, 10001):
    if prime_list[i]:
        j = 1
        while True:
            try:
                j += 1
                prime_list[i*j] = False
            except IndexError:
                break

m, n = map(int, input().split())
for i in range(m):
    marks = [int(j) for j in input().split()]
    nm = len(marks)
    mark_sum = 0
    for mark in marks:
        if abs(mark**0.5 % 1 - 0) < 0.00000001:
            if prime_list[int(mark**0.5)]:
                mark_sum += mark
    if mark_sum == 0:
        print(0)
    else:
        print("%.2f" % (mark_sum/nm))
```

状态: Accepted

源代

```
prime_list = [True for i in range(10001)]
prime_list[1] = False

for i in range(1, 10001):
    if prime_list[i]:
        j = 1
        while True:
            try:
                j += 1
                prime_list[i*j] = False
            except IndexError:
                break

m, n = map(int, input().split())
for i in range(m):
    marks = [int(j) for j in input().split()]
    nm = len(marks)
    mark_sum = 0
    for mark in marks:
        if abs(mark**0.5 % 1 - 0) < 0.00000001:
            if prime_list[int(mark**0.5)]:
                mark_sum += mark
    if mark_sum == 0:
        print(0)
    else:
        print("%.2f" % (mark_sum/nm))
```

基本信息

#:	43010605
题目:	18176
提交人:	23n2300011538
内存:	3864kB
时间:	74ms
语言:	Python3
提交时间:	2023-12-08 16:48:21

2. 学习总结和收获

都是以前做过的or（第一题）最近放在每日选做里面的

比较有空，故不定时做一些每日选做，暂时对我来说难度不大

数算pre每日选昨也会做，但是要现学图or树的写法；这是chatgpt非常擅长的方面，故常常求助于chatgpt