

Assignment #2: 编程练习

Updated 0953 GMT+8 Feb 24, 2024

2024 spring, Compiled by 数学科学学院 王镜廷 2300010724

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

操作系统: Windows11 专业版

Python编程环境: VSCode 1.86.2, with extension Python and python version 3.12.2

1. 题目

27653: Fraction类

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

用时约10分钟

思路:

创建Fraction类并写出对应函数

代码

```
def getGcd(m, n) :
    if m == 0 or n == 0 :
        return m + n
    return getGcd(n, m % n)

class Fraction :
    num = 1
    den = 0

    def __init__(self, top, bottom) :
        self.num = top
        self.den = bottom

    def __eq__(self, other) :
        return self.num * other.den == self.den * other.num

    def __str__(self) :
        return str(self.num)+"/"+str(self.den)

    def __add__(self, other) :
        return Fraction(self.num * other.den + self.den * other.num, self.den * other.den)

    def simp(self) :
        return Fraction(self.num // getGcd(self.num, self.den), self.den // getGcd(self.num, self.den))

a = list(map(int, input().split()))
x = Fraction(a[0], a[1])
y = Fraction(a[2], a[3])
print((x + y).simp())
```



代码运行截图（至少包含有"Accepted"）

状态: Accepted

源代码

```
def getGcd(m, n) :  
    if m == 0 or n == 0 :  
        return m + n  
    return getGcd(n, m % n)  
class Fraction :  
    num = 1  
    den = 0  
    def __init__(self, top, bottom) :  
        self.num = top  
        self.den = bottom  
  
    def __eq__(self, other) :  
        return self.num * other.den == self.den * other.num  
  
    def __str__(self) :  
        return str(self.num) + "/" + str(self.den)  
  
    def __add__(self, other) :  
        return Fraction(self.num * other.den + self.den * other.num, self.den * other.den)  
  
    def simp(self) :  
        return Fraction(self.num // getGcd(self.num, self.den), self.den // getGcd(self.num, self.den))  
  
a = list(map(int, input().split()))  
x = Fraction(a[0], a[1])  
y = Fraction(a[2], a[3])  
print((x + y).simp())
```

基本信息

#: 43942681

题目: 27653

提交人: 2300010724 王镜廷

内存: 3540kB

时间: 20ms

语言: Python3

提交时间: 2024-02-20 21:18:15

04110: 圣诞老人的礼物-Santa Clau's Gifts

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

用时约10分钟

思路:

对糖果按照价值/重量排序, 之后从大往小依次取

代码

```
class Candy :
    val = 0
    vol = 0
    def __init__(self, val, vol) :
        self.val = val
        self.vol = vol
    def __lt__(self, other) :
        return self.val * other.vol < self.vol * other.val

n, w = map(int, input().split())
listGift = []
res = 0
VOL = w
for i in range(n) :
    u, v = map(int, input().split())
    listGift.append(Candy(u, v))
listGift = sorted(listGift, reverse = True)
for candy in listGift :
    if candy.vol <= VOL :
        res += candy.val
        VOL -= candy.vol
    else :
        res += (candy.val / candy.vol) * VOL
        break
print(f"{res:.1f}")
```

代码运行截图（至少包含有"Accepted"）

状态: Accepted

源代码

```
class Candy :
    val = 0
    vol = 0
    def __init__(self, val, vol) :
        self.val = val
        self.vol = vol
    def __lt__(self, other) :
        return self.val * other.vol < self.vol * other.val

n, w = map(int, input().split())
listGift = []
res = 0
VOL = w
for i in range(n) :
    u, v = map(int, input().split())
    listGift.append(Candy(u, v))
listGift = sorted(listGift, reverse = True)
for candy in listGift :
    if candy.vol <= VOL :
        res += candy.val
        VOL -= candy.vol
    else :
        res += (candy.val / candy.vol) * VOL
        break
print(f"{res:.1f}")
```

基本信息

#: 43981229
题目: 04110
提交人: 2300010724 王镜廷
内存: 3612kB
时间: 21ms
语言: Python3
提交时间: 2024-02-24 20:30:31

18182: 打怪兽

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

用时约12分钟

思路:

将技能先按照时刻从小到大排序，时刻相同的按伤害从大到小排，之后依次使用技能，每个技能使用时若该时刻已经施放过m个技能就跳过。

代码

```
class Skill :
    def __init__(self, time, hit) :
        self.time = time
        self.hit = hit
    def __lt__(self, other) :
        return (self.time < other.time) or ((self.time == other.time) and (self.hit > other.hit))
    def __str__(self) :
        return "(" + str(self.time) + ", " + str(self.hit) + ")"

nCases = int(input())
listSkill = []
for Case in range(nCases) :
    n, m, b = map(int, input().split())
    listSkill = []
    for i in range(n) :
        u, v = map(int, input().split())
        listSkill.append(Skill(u, v))
    listSkill = sorted(listSkill)
    Time = 0
    cnt = 0
    hp = b
    for skill in listSkill :
        if skill.time != Time :
            Time = skill.time
            cnt = 1
            hp -= skill.hit
        else :
            if cnt < m :
                cnt += 1
                hp -= skill.hit
            else :
                continue
        if hp <= 0 :
            print(Time)
            break
    if hp > 0 :
        print("alive")
```



代码运行截图（AC代码截图，至少包含有"Accepted"）

状态: Accepted

源代码

```
class Skill :
    def __init__(self, time, hit) :
        self.time = time
        self.hit = hit
    def __lt__(self, other) :
        return (self.time < other.time) or ((self.time == other.time) and self.hit < other.hit)
    def __str__(self) :
        return "(" + str(self.time) + ", " + str(self.hit) + ")"

nCases = int(input())
listSkill = []
for Case in range(nCases) :
    n, m, b = map(int, input().split())
    listSkill = []
    for i in range(n) :
        u, v = map(int, input().split())
        listSkill.append(Skill(u, v))
    listSkill = sorted(listSkill)
    Time = 0
    cnt = 0
    hp = b
    for skill in listSkill :
        if skill.time != Time :
            Time = skill.time
            cnt = 1
            hp -= skill.hit
        else :
            if cnt < m :
                cnt += 1
                hp -= skill.hit
            else :
                continue
        if hp <= 0 :
            print(Time)
            break
    if hp > 0 :
        print("alive")
```

基本信息

#: 43981324
题目: 18182
提交人: 2300010724 王镜廷
内存: 3772kB
时间: 100ms
语言: Python3
提交时间: 2024-02-24 20:46:29

230B. T-primes

binary search/implementation/math/number theory, 1300,

<http://codeforces.com/problemset/problem/230/B>

用时约5分钟

思路:

先判断每个数是否为平方数，若是，再将每个数开平方根并判断平方根是否为质数。

代码

```
import math

isprime = [1] * 1000005
isprime[0] = 0
isprime[1] = 0
for i in range(2, 1000005) :
    if isprime[i] == 1 :
        for j in range(i * i, 1000005, i) :
            isprime[j] = 0
n = int(input())
listNum = list(map(int, input().split()))
for num in listNum :
    u = int(math.sqrt(num))
    if u * u == num and isprime[u] == 1 :
        print("YES")
    else :
        print("NO")
```

代码运行截图（AC代码截图，至少包含有"Accepted"）

By wangjingting, contest: Codeforces Round 142 (Div. 2), problem: (B) T-primes, **Accepted**, #, [Copy](#).

```
import math

isprime = [1] * 1000005
isprime[0] = 0
isprime[1] = 0
for i in range(2, 1000005) :
    if isprime[i] == 1 :
        for j in range(i * i, 1000005, i) :
            isprime[j] = 0
n = int(input())
listNum = list(map(int, input().split()))
for num in listNum :
    u = int(math.sqrt(num))
    if u * u == num and isprime[u] == 1 :
        print("YES")
    else :
        print("NO")
```

1364A. XXXXX

brute force/data structures/number theory/two pointers, 1200,

<https://codeforces.com/problemset/problem/1364/A>

约8分钟

思路：

取前缀和后，若总和不为x的倍数则直接输出n，若总和为x的倍数，取出最左端前缀和不被x整除的下标，最右端前缀和不被x整除的下标r，则答案为n-l与r中较大者

代码

```
nCases = int(input())
for case in range(nCases) :
    n, x = map(int, input().split())
    listNum = list(map(int, input().split()))
    Sum = [0] * (n + 1)
    for i in range(n) :
        Sum[i + 1] = Sum[i] + listNum[i]
    l = 0
    for i in range(1, n + 1) :
        if (Sum[i]) % x != 0 :
            l = i
            break
    if l == 0 :
        print("-1")
        continue
    r = 0
    for i in range(1, n + 1) :
        if (Sum[n + 1 - i] - Sum[n]) % x != 0 :
            r = n + 1 - i
            break
    if (Sum[n]) % x != 0 :
        print(f"{n}")
        continue
    else :
        print(f"{max(r, n - l)}")
```

代码运行截图（AC代码截图，至少包含有"Accepted"）

By [wangjingting](#), contest: Codeforces Round 649 (Div. 2), problem: (A) XXXXX, **Accepted**, #, [Copy](#)

```
nCases = int(input())
for case in range(nCases) :
    n, x = map(int, input().split())
    listNum = list(map(int, input().split()))
    Sum = [0] * (n + 1)
    for i in range(n) :
        Sum[i + 1] = Sum[i] + listNum[i]
    l = 0
    for i in range(1, n + 1) :
        if (Sum[i]) % x != 0 :
            l = i
            break
    if l == 0 :
        print("-1")
        continue
    r = 0
    for i in range(1, n + 1) :
        if (Sum[n + 1 - i] - Sum[n]) % x != 0 :
            r = n + 1 - i
            break
    if (Sum[n]) % x != 0 :
        print(f"{n}")
        continue
    else :
        print(f"{max(r, n - l)}")
```

18176: 2050年成绩计算

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

用时约10分钟

思路:

依次判断每个成绩是否为质数的平方

代码

```
import math

isprime = [1] * 20005
isprime[0] = 0
isprime[1] = 0
for i in range(2, 20005) :
    if isprime[i] :
        for j in range(i * i, 20005, i) :
            isprime[j] = 0
s = input().split()
m = int(s[0])
tot = 0
cnt = 0
for i in range(m) :
    s = input().split()
    tot = 0
    cnt = 0
    flag = 0
    for item in s :
        u = int(item)
        v = int(math.sqrt(u))
        if v * v == u and isprime[v] == 1 :
            tot += u
            flag = 1
    cnt += 1
if flag == 0 :
    print(0)
else :
    print(f"{tot / cnt :.2f}")
```

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

isprime = [1] * 20005
isprime[0] = 0
isprime[1] = 0
for i in range(2, 20005) :
    if isprime[i] :
        for j in range(i * i, 20005, i) :
            isprime[j] = 0
s = input().split()
m = int(s[0])
tot = 0
cnt = 0
for i in range(m) :
    s = input().split()
    tot = 0
    cnt = 0
    flag = 0
    for item in s :
        u = int(item)
        v = int(math.sqrt(u))
        if v * v == u and isprime[v] == 1 :
            tot += u
            flag = 1
    cnt += 1
    if flag == 0 :
        print(0)
    else :
        print(f"{tot / cnt :.2f}")
```

基本信息

: 43975850

题目: 18176

提交人: 2300010724 王镜廷

内存: 4168kB

时间: 71ms

语言: Python3

提交时间: 2024-02-24 08:48:52

2. 学习总结和收获

本周完成了每日选做的题目，上机课上Virtual Practice了Codeforces Round 926 (Div. 2)，同时学到了python里面类的基础操作。