

Assignment #5: 20251009 cs101 Mock Exam寒露第二天

Updated 1651 GMT+8 Oct 9, 2025

2025 fall, Compiled by 物理学院 李欣珂

说明:

1. 解题与记录:

对于每一个题目，请提供其解题思路（可选），并附上使用Python或C++编写的源代码（确保已在OpenJudge, Codeforces, LeetCode等平台上获得Accepted）。请将这些信息连同显示“Accepted”的截图一起填写到下方的作业模板中。（推荐使用Typora <https://typoraio.cn> 进行编辑，当然你也可以选择Word。）无论题目是否已通过，请标明每个题目大致花费的时间。

2. 提交安排: **提交时，请首先上传PDF格式的文件，并将.md或.doc格式的文件作为附件上传至右侧的“作业评论”区。确保你的Canvas账户有一个清晰可见的本人头像，提交的文件为PDF格式，并且“作业评论”区包含上传的.md或.doc附件。

3. **延迟提交: **如果你预计无法在截止日期前提交作业，请提前告知具体原因。这有助于我们了解情况并可能为你提供适当的延期或其他帮助。

请按照上述指导认真准备和提交作业，以保证顺利完成课程要求。

1. 题目

E29895: 分解因数

implementation, <http://cs101.openjudge.cn/practice/29895/>

思路：一开始超时，后面意识到事实上不必遍历所有数，小的数判断了后大的数自然直接得到结果，优化后解决

代码

```
n=int(input())
p=0
for i in range(2,int(n**0.5)+1):
    if int(n/i)==n/i:
        p=i
        break
print(int(n/p))
```

代码运行截图 (至少包含有"Accepted")

#50286266提交状态

查看提交统计提问

状态: Accepted

源代码

```
n=int(input())
p=0
for i in range(2,int(n**0.5)+1):
    if int(n/i)==n/i:
        p=i
        break
print(int(n/p))
```

基本信息

#: 50286266

题目: 29895

提交人: 25n2500011474

内存: 3604kB

时间: 18ms

语言: Python3

提交时间: 2025-10-10 09:16:29

©2002-2022 POJ 京ICP备20010980号-1

English帮助关于

E29940: 机器猫斗恶龙

greedy, <http://cs101.openjudge.cn/practice/29940/>

思路: 个人认为比较简单的语法题

代码

```
n=int(input())
a=list(map(int,input().split()))
A=0
b=[]
for i in range(n):
    A+=a[i]
    b.append(A)
print(-min(b)+1)
```

代码运行截图 (至少包含有"Accepted")

#50286296提交状态

查看提交统计提问

状态: Accepted

源代码

```
n=int(input())
a=list(map(int,input().split()))
A=0
b=[]
for i in range(n):
    A+=a[i]
    b.append(A)
print(-min(b)+1)
```

基本信息

#: 50286296

题目: 29940

提交人: 25n2500011474

内存: 14884kB

时间: 55ms

语言: Python3

提交时间: 2025-10-10 09:18:57

©2002-2022 POJ 京ICP备20010980号-1

English帮助关于

M29917: 牛顿迭代法

implementation, <http://cs101.openjudge.cn/practice/29917/>

思路：花了点时间去回忆牛顿法如何迭代（感觉题目表述事实上没有全盘告诉你牛顿法，没见过的同学需要自己思考其中原理？），想明白了自然也就成了语法题

代码

```
import sys
for line in sys.stdin:
    n = float(line.strip())
    A = 0
    x = 1
    while True:
        if abs((x ** 2 - n) / (2 * x)) <= 0.000001:
            break
        else:
            x = x - (x ** 2 - n) / (2 * x)
            A += 1
    print(f"{A+1} {x:.2f}")
```

代码运行截图 (至少包含有"Accepted")

#50286303提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: [Accepted](#)

源代码

```
import sys
for line in sys.stdin:
    n = float(line.strip())
    A = 0
    x = 1
    while True:
        if abs((x ** 2 - n) / (2 * x)) <= 0.000001:
            break
        else:
            x = x - (x ** 2 - n) / (2 * x)
            A += 1
    print(f"{A+1} {x:.2f}")
```

基本信息

#: 50286303
 题目: 29917
 提交人: 25n2500011474
 内存: 3488kB
 时间: 21ms
 语言: Python3
 提交时间: 2025-10-10 09:19:46

©2002-2022 POJ 京ICP备20010980号-1

[English](#) [帮助](#) [关于](#)

M29949: 贪婪的哥布林

greedy, <http://cs101.openjudge.cn/practice/29949/>

思路：较为简单的贪心，考试的时候还好把这个做了

代码

```
N,M=map(int,input().split())
A=[]
for _ in range(N):
    v,w=map(int,input().split())
    A.append((v,w,v/w))
A.sort(reverse=True,key=lambda x:x[2])
value=0
```

```

weight=M
for (v,w,rate) in A:
    if w<=weight:
        weight-=w
        value+=v
    else:
        value+=rate*weight
        break
print(f"{value:.2f}")

```

代码运行截图 (至少包含有"Accepted")

#50286620提交状态

查看

提交

统计

提问

状态: Accepted

源代码

```

N,M=map(int,input().split())
A=[]
for _ in range(N):
    v,w=map(int,input().split())
    A.append((v,w,v/w))
A.sort(reverse=True,key=lambda x:x[2])
value=0
weight=M
for (v,w,rate) in A:
    if w<=weight:
        weight-=w
        value+=v
    else:
        value+=rate*weight
        break
print(f"{value:.2f}")

```

基本信息

#: 50286620

题目: 29949

提交人: 25n2500011474

内存: 3632kB

时间: 19ms

语言: Python3

提交时间: 2025-10-10 09:47:06

©2002-2022 POJ 京ICP备20010980号-1

English 帮助 关于

M29918: 求亲和数

implementation, <http://cs101.openjudge.cn/practice/29918/>

思路：考试时重复计算数的亲和数导致做法超时了，但是一直没反应过来被卡了好久，自己下去写稍微优化了一点点（虽然没好到哪去，但是能过），采取预先计算所有数的亲和数并存表的方法；不过自己经验不足，考试时程序事实上能本地得到 $n=100000$ 以内的所有亲和数，再直接做，虽然投机取巧，但也是一种考试技巧，下次月考如果遇到做不出来的情况也会想巧办法。此外，ai给出了一个计算亲和数的更加简便的做法：

```

n = int(input())

# 预计算所有数的真因数之和
sum_of_divisors = [0] * (n + 1)
for i in range(1, n + 1):
    for j in range(i * 2, n + 1, i):
        sum_of_divisors[j] += i

# 查找亲和数对
pairs = []
for i in range(2, n + 1):

```

```

j = sum_of_divisors[i]
if j > i and j <= n and sum_of_divisors[j] == i:
    pairs.append((i, j))

# 输出结果
for a, b in pairs:
    print(a, b)

```

这个做法不需要计算出每个数的因数，而是把每个因数加到其倍数对应的列表上，时间复杂度大大降低

代码

```

import math
def add(a):
    A=1
    for i in range(2,int(math.sqrt(a))+1):
        if a % i == 0:
            if i == int(a/i):
                A += i
            else:
                A += i + int(a/i)
    return A
n=int(input())
B=[]
adds=[0]*(n+1)
for i in range(1,n+1):
    adds[i]=add(i)
for i in range(1,n+1):
    if adds[i]<=n and i<adds[i]:
        if adds[adds[i]]==i:
            B.append((i,adds[i]))
B.sort(key=lambda x:x[0],reverse=False)
for i in B:
    print(int(i[0]),int(i[1]))

```

代码运行截图 (至少包含有"Accepted")

#50291537提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: Accepted

源代码

```
import math
def add(a):
    A=1
    for i in range(2,int(math.sqrt(a))+1):
        if a % i == 0:
            if i == int(a/i):
                A += i
            else:
                A += i + int(a/i)
    return A
n=int(input())
B=[]
adds=[0]*(n+1)
for i in range(1,n+1):
    adds[i]=add(i)
for i in range(1,n+1):
    if adds[i]<=n and i<adds[i]:
        if adds[adds[i]]==i:
            B.append((i,adds[i]))
B.sort(key=lambda x:x[0],reverse=False)
for i in B:
    print(int(i[0]),int(i[1]))
```

基本信息

#: 50291537
题目: 29918
提交人: 25n2500011474
内存: 7028kB
时间: 1056ms
语言: Python3
提交时间: 2025-10-10 16:16:20

©2002-2022 POJ 京ICP备20010980号-1

[English](#) [帮助](#) [关于](#)

T29947:校门外的树又来了 (选做)

<http://cs101.openjudge.cn/practice/29947/>

思路：一开始做的时候建立了一个长度为L的列表，结果导致MLE；于是意识到应该直接处理区间，采取区间合并的思路，将所有区间按起点排序，并依次判断能否合并，并对最后的区间计算长度，这样一来时间和空间复杂度都很小了。

代码

```
L,M=map(int,input().split())
sta=[]
A=0
for i in range(M):
    a,b=map(int,input().split())
    sta.append((a,b))
sta.sort()
for i in range(M):
    if i<M-1:
        if sta[i][1]>=sta[i+1][0]:
            sta[i+1]=(sta[i][0],max(sta[i][1],sta[i+1][1]))
            sta[i]=(0, 0)
            A+=sta[i][1]-sta[i][0]
        else:
            A+=sta[i][1]-sta[i][0]+1
    else:
        A += sta[i][1] - sta[i][0]+1
print(L-A+1)
```

代码运行截图 (至少包含有"Accepted")

#50294366提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: Accepted

源代码

```
L,M=map(int,input().split())
sta=[]
A=0
for i in range(M):
    a,b=map(int,input().split())
    sta.append((a,b))
sta.sort()
for i in range(M):
    if i<M-1:
        if sta[i][1]>=sta[i+1][0]:
            sta[i+1]=(sta[i][0],max(sta[i][1],sta[i+1][1]))
            sta[i]=(0, 0)
            A+=sta[i][1]-sta[i][0]
        else:
            A+=sta[i][1]-sta[i][0]+1
    else:
        A += sta[i][1] - sta[i][0]+1
print(L-A+1)
```

基本信息

#: 50294366
题目: 29947
提交人: 25n2500011474
内存: 3660kB
时间: 19ms
语言: Python3
提交时间: 2025-10-10 17:21:59

2. 学习总结和收获

这次考试ac4，亲和数一题超时卡了很久导致没有时间去做了T题。个人认为原因是对于代码的优化等不太熟悉。自己最近要学习一下对于各种复杂度的优化办法，并学习一些更加先进的算法。