Assignment #3: March月考

Updated 1537 GMT+8 March 6, 2024

2024 spring, Complied by 尹柚鑫 光华管理学院 2100015878

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）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：windows11

Python编程环境：jupter notebook

1. 题目

02945: 拦截导弹

http://cs101.openjudge.cn/practice/02945/

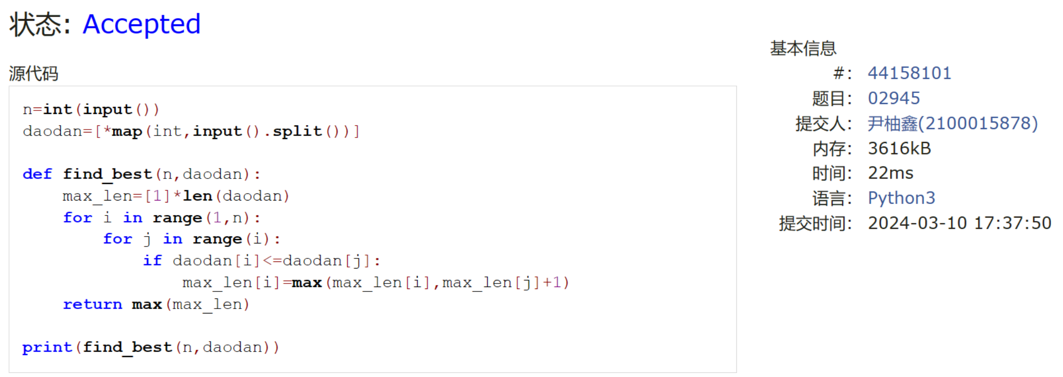
耗时：30mins

思路：动态规划，建立一个列表，每个位置的数代表如果拦截对应位置的导弹的话（并且是最后一个），能够拦截几个，可以从前面的某个数得到

代码

```python
# 
n=int(input())
daodan=[*map(int,input().split())]

def find_best(n,daodan):
    max_len=[1]*len(daodan)
    for i in range(1,n):
        for j in range(i):
            if daodan[i]<=daodan[j]:
                max_len[i]=max(max_len[i],max_len[j]+1)
    return max(max_len)

print(find_best(n,daodan))
```

代码运行截图 ==（至少包含有"Accepted"）==



状态: Accepted

源代码

```python
n=int(input())
daodan=[*map(int,input().split())]

def find_best(n,daodan):
    max_len=[1]*len(daodan)
    for i in range(1,n):
        for j in range(i):
            if daodan[i]<=daodan[j]:
                max_len[i]=max(max_len[i],max_len[j]+1)
    return max(max_len)

print(find_best(n,daodan))
```

基本信息

#:	44158101
题目:	02945
提交人:	尹柚鑫(2100015878)
内存:	3616kB
时间:	22ms
语言:	Python3
提交时间:	2024-03-10 17:37:50

04147:汉诺塔问题(Tower of Hanoi)

http://cs101.openjudge.cn/practice/04147

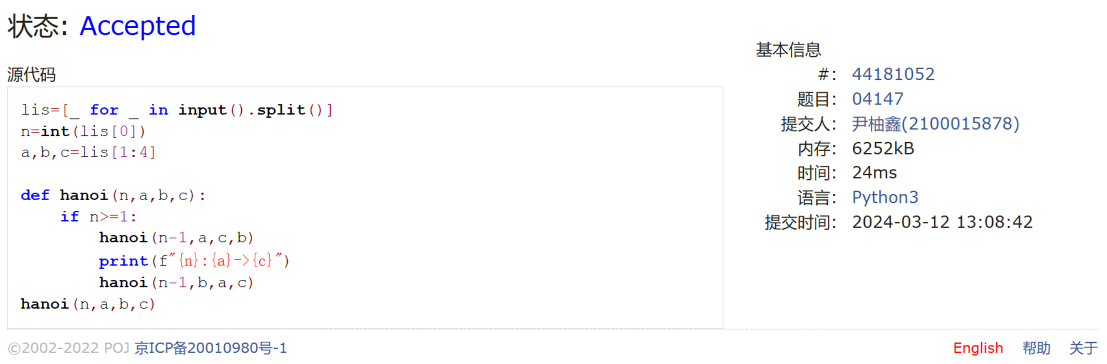
耗时：10mins

思路：经典递归

代码

```python
# 
lis=[_ for _ in input().split()]
n=int(lis[0])
a,b,c=lis[1:4]

def hanoi(n,a,b,c):
    if n>=1:
        hanoi(n-1,a,c,b)
        print(f"{n}:{a}->{c}")
        hanoi(n-1,b,a,c)
hanoi(n,a,b,c)
```

代码运行截图 ==（至少包含有"Accepted"）==



状态: Accepted

源代码

```
lis=[_ for _ in input().split()]
n=int(lis[0])
a,b,c=lis[1:4]

def hanoi(n,a,b,c):
    if n>=1:
        hanoi(n-1,a,c,b)
        print(f"{n}:{a}->{c}")
        hanoi(n-1,b,a,c)
hanoi(n,a,b,c)
```

基本信息

#:	44181052
题目:	04147
提交人:	尹柚鑫(2100015878)
内存:	6252kB
时间:	24ms
语言:	Python3
提交时间:	2024-03-12 13:08:42

©2002-2022 POJ 京ICP备20010980号-1

English 帮助 关于

03253: 约瑟夫问题No.2

http://cs101.openjudge.cn/practice/03253

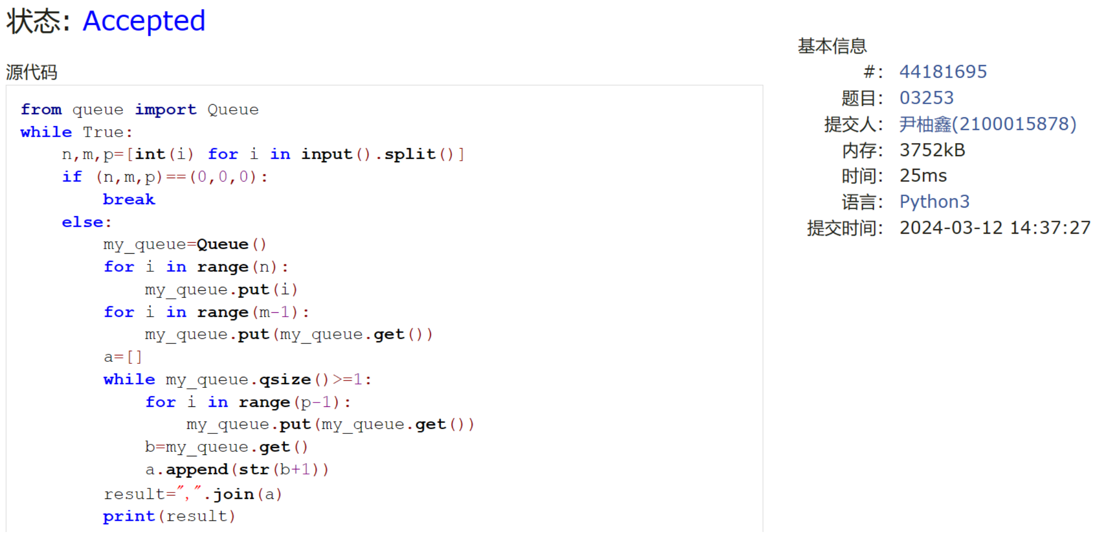
耗时：20mins

思路：使用队列的思想，非常适合解决这类同余有关的题目

代码

```python
#
from queue import Queue
while True:
    n,m,p=[int(i) for i in input().split()]
    if (n,m,p)==(0,0,0):
        break
    else:
        my_queue=Queue()
        for i in range(n):
            my_queue.put(i)
        for i in range(m-1):
            my_queue.put(my_queue.get())
        a=[]
        while my_queue.qsize()>=1:
            for i in range(p-1):
                my_queue.put(my_queue.get())
            b=my_queue.get()
            a.append(str(b+1))
        result=",".join(a)
        print(result)
```

代码运行截图 ==（AC代码截图，至少包含有"Accepted"）==



状态: Accepted

源代码

```
from queue import Queue
while True:
    n,m,p=[int(i) for i in input().split()]
    if (n,m,p)==(0,0,0):
        break
    else:
        my_queue=Queue()
        for i in range(n):
            my_queue.put(i)
        for i in range(m-1):
            my_queue.put(my_queue.get())
        a=[]
        while my_queue.qsize()>=1:
            for i in range(p-1):
                my_queue.put(my_queue.get())
            b=my_queue.get()
            a.append(str(b+1))
        result=",".join(a)
        print(result)
```

基本信息

#:	44181695
题目:	03253
提交人:	尹柚鑫(2100015878)
内存:	3752kB
时间:	25ms
语言:	Python3
提交时间:	2024-03-12 14:37:27

21554:排队做实验 (greedy)v0.2

http://cs101.openjudge.cn/practice/21554

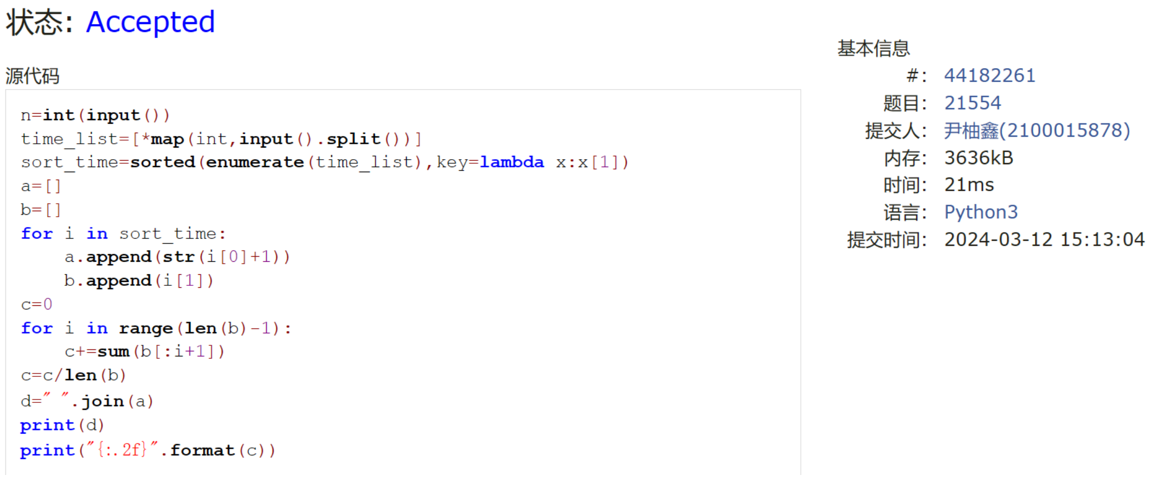
耗时：10mins

思路：对所需时间从小到大排序即可

代码

```
#
n=int(input())
time_list=[*map(int,input().split())]
sort_time=sorted(enumerate(time_list),key=lambda x:x[1])
a=[]
b=[]
for i in sort_time:
    a.append(str(i[0]+1))
    b.append(i[1])
c=0
for i in range(len(b)-1):
    c+=sum(b[:i+1])
c=c/len(b)
d=" ".join(a)
print(d)
print("{:.2f}".format(c))
```

代码运行截图 ==（AC代码截图，至少包含有"Accepted"）==


状态: Accepted

源代码

```
n=int(input())
time_list=[*map(int,input().split())]
sort_time=sorted(enumerate(time_list),key=lambda x:x[1])
a=[]
b=[]
for i in sort_time:
    a.append(str(i[0]+1))
    b.append(i[1])
c=0
for i in range(len(b)-1):
    c+=sum(b[:i+1])
c=c/len(b)
d=" ".join(a)
print(d)
print("{:.2f}".format(c))
```

基本信息

#: 44182261

题目: 21554

提交人: 尹柚鑫(2100015878)

内存: 3636kB

时间: 21ms

语言: Python3

提交时间: 2024-03-12 15:13:04

19963:买学区房

http://cs101.openjudge.cn/practice/19963

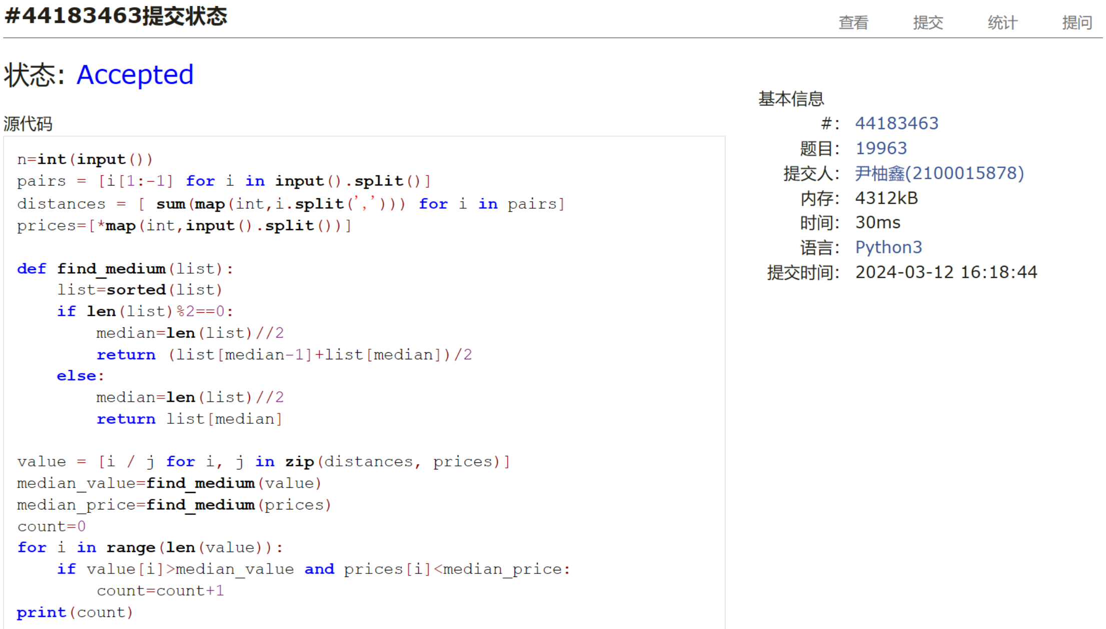
耗时：10mins

思路：比较简单，排序求中位数就行

代码

```python
#
n=int(input())
pairs = [i[1:-1] for i in input().split()]
distances = [ sum(map(int,i.split(','))) for i in pairs]
prices=[*map(int,input().split())]

def find_medium(list):
    list=sorted(list)
    if len(list)%2==0:
        median=len(list)//2
        return (list[median-1]+list[median])/2
    else:
        median=len(list)//2
        return list[median]

value = [i / j for i, j in zip(distances, prices)]
median_value=find_medium(value)
median_price=find_medium(prices)
count=0
for i in range(len(value)):
    if value[i]>median_value and prices[i]<median_price:
        count=count+1
print(count)
```

代码运行截图 ==（AC代码截图，至少包含有"Accepted"）==



#44183463提交状态

状态: Accepted

源代码

```
n=int(input())
pairs = [i[1:-1] for i in input().split()]
distances = [ sum(map(int,i.split(','))) for i in pairs]
prices=[*map(int,input().split())]

def find_medium(list):
    list=sorted(list)
    if len(list)%2==0:
        median=len(list)//2
        return (list[median-1]+list[median])/2
    else:
        median=len(list)//2
        return list[median]

value = [i / j for i, j in zip(distances, prices)]
median_value=find_medium(value)
median_price=find_medium(prices)
count=0
for i in range(len(value)):
    if value[i]>median_value and prices[i]<median_price:
        count=count+1
print(count)
```

基本信息

#: 44183463
题目: 19963
提交人: 尹柚鑫(2100015878)
内存: 4312kB
时间: 30ms
语言: Python3
提交时间: 2024-03-12 16:18:44

27300: 模型整理

http://cs101.openjudge.cn/practice/27300

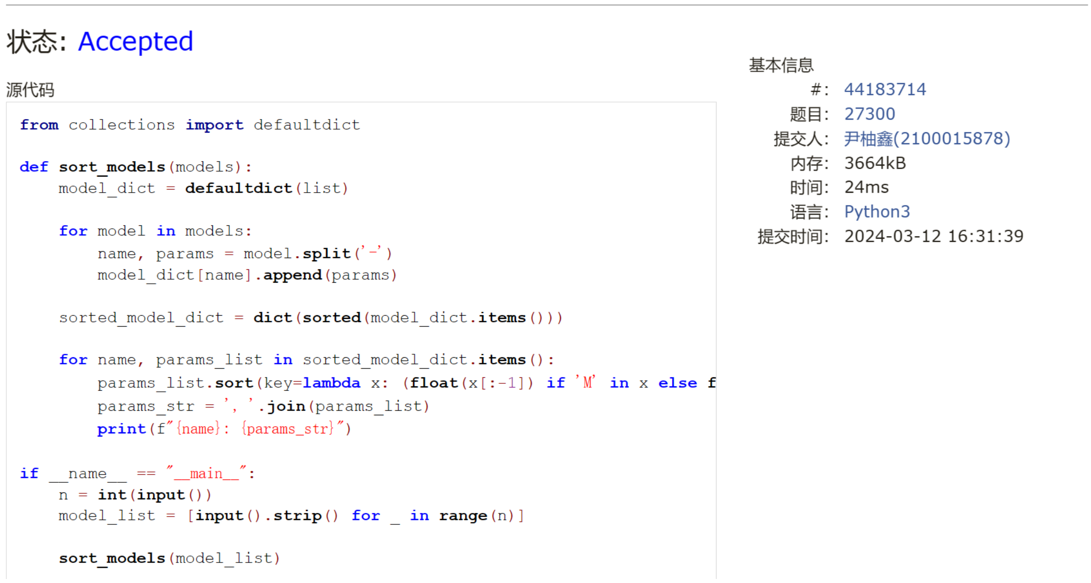
耗时：20mins

思路：

代码

```python
# 
from collections import defaultdict

def sort_models(models):
    model_dict = defaultdict(list)

    for model in models:
        name, params = model.split('-')
        model_dict[name].append(params)

    sorted_model_dict = dict(sorted(model_dict.items()))

    for name, params_list in sorted_model_dict.items():
        params_list.sort(key=lambda x: (float(x[:-1]) if 'M' in x else
float(x[:-1]) * 1000))
        params_str = ', '.join(params_list)
        print(f"{name}: {params_str}")

if __name__ == "__main__":
    n = int(input())
    model_list = [input().strip() for _ in range(n)]

    sort_models(model_list)
```

代码运行截图 ==（AC代码截图，至少包含有"Accepted"）==



状态: Accepted

源代码

```
from collections import defaultdict

def sort_models(models):
    model_dict = defaultdict(list)

    for model in models:
        name, params = model.split('-')
        model_dict[name].append(params)

    sorted_model_dict = dict(sorted(model_dict.items()))

    for name, params_list in sorted_model_dict.items():
        params_list.sort(key=lambda x: (float(x[:-1]) if 'M' in x else f
        params_str = ', '.join(params_list)
        print(f"{name}: {params_str}")

if __name__ == "__main__":
    n = int(input())
    model_list = [input().strip() for _ in range(n)]

    sort_models(model_list)
```

基本信息

#:	44183714
题目:	27300
提交人:	尹柚鑫(2100015878)
内存:	3664kB
时间:	24ms
语言:	Python3
提交时间:	2024-03-12 16:31:39

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。==

1.在Python中，`*`（星号）被用作解包操作符。在你提供的代码中，`*map(int, input().split())` 的作用是将输入的一行字符串分割成多个部分，并将每个部分转换为整数，最终将这些整数解包成一个列表。

daodan=[*map(int,input().split())]

2.熟悉python自带的队列类的函数

3.关于对 350M, 1.3B, 175B排序

如果没有认为M后缀一定比B后缀小，就要转化单位，下面这行代码写的很利落

params_list.sort(key=lambda x: (float(x[:-1]) if 'M' in x else float(x[:-1]) * 1000))

这一题中M后缀一定比B后缀小，可以像下面这样写：建元组，第二个数是0或者1表示标记，再对两个标准排序

def custom_sort(value):

number, suffix = float(value[:-1]), value[-1]

return (number, 0) if suffix == 'M' else (number, 1)

chatgpt写的python真是在小小的语法上非常灵活，这个return还能跟if else