Assignment #5: Greedy穷举Implementation

Updated 1939 GMT+8 Oct 21, 2024

2024 fall, Complied by 颜鼎堃 工学院

说明：

1）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。

1. 题目

04148: 生理周期

brute force, http://cs101.openjudge.cn/practice/04148

思路：

- 中国剩余定理
- 顺便学习了一下扩展欧几里得算法（代码中注释部分）

代码：

```
# def exgcd(a, b):
#       if b == 0:
#               x = 1
#               y = 0
#               return x, y
#       x1, y1 = exgcd(b, a%b)
#       x = y1
#       y = x1 - a//b*y1
#       return x, y

def main():
        for cnt in range(1, int(1e9)):
                p, e, i, d = map(int, input().split())
                if i == -1:
                        break
                x = (1288*i - 6831*e+ 5544*p) % 21252 - d
```

```
                print(f"Case {cnt}: the next triple peak occurs in {(x-1) %
21252 + 1} days.")

if __name__ == '__main__':
        main()
```

代码运行截图 （至少包含有"Accepted"）

18211: 军备竞赛

greedy, two pointers, http://cs101.openjudge.cn/practice/18211

思路：

- 双指针，一边负责买一边负责卖，每一轮循环中只要买了就可以卖

代码：

```
p = int(input())
a = sorted(list(map(int, input().split())))
fwd, bwd = 0, len(a) - 1
cnt = 0
sell = False
while True:
        while p >= a[fwd]:
                p -= a[fwd]
                cnt += 1
                sell = True
                fwd += 1
                if bwd < fwd:
```

```
                        sell = False
                        break
        if sell:
                cnt -= 1
                p += a[bwd]
                bwd -= 1
                sell = False
        else:
                break
print(cnt)
```

代码运行截图 （至少包含有"Accepted"）

21554: 排队做实验

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

思路：

- 排序不等式经典例题
- 输出的时候注意格式，用空格分隔确实烦人

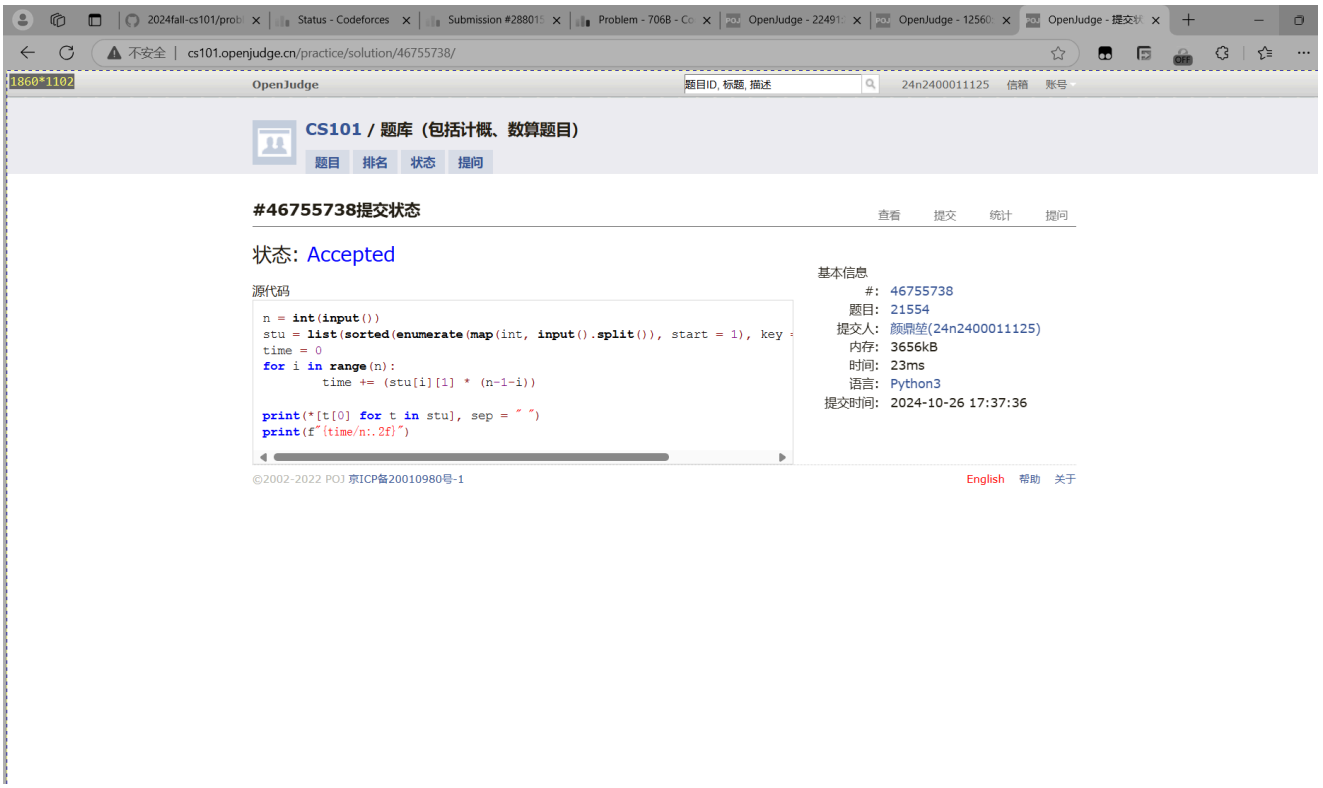
代码：

```
n = int(input())
stu = list(sorted(enumerate(map(int, input().split()), start = 1), key =
lambda t: t[1]))
time = 0
for i in range(n):
```

```
        time += (stu[i][1] * (n-1-i))

print(*[t[0] for t in stu], sep = " ")
print(f"{time/n:.2f}")
```

代码运行截图 （至少包含有"Accepted"）



01008: Maya Calendar

implementation, http://cs101.openjudge.cn/practice/01008/

思路：

- 数清楚天数
- 输出的时候把`n`再输出一遍
- 学会了`zip`函数将列表变成元组，进而变成字典
  代码：

```
n = int(input())
print(n)
Hmonths = {'pop': 1, 'no': 2, 'zip': 3, 'zotz': 4, 'tzec': 5, 'xul': 6,
'yoxkin': 7, 'mol': 8, 'chen': 9, 'yax': 10, 'zac': 11, 'ceh': 12, 'mac':
13, 'kankin': 14, 'muan': 15, 'pax': 16, 'koyab': 17, 'cumhu': 18, 'uayet':
19}
Tdays = {0: 'imix', 1: 'ik', 2: 'akbal', 3: 'kan', 4: 'chicchan', 5: 'cimi',
6: 'manik', 7: 'lamat', 8: 'muluk', 9: 'ok', 10: 'chuen', 11: 'eb', 12:
'ben', 13: 'ix', 14: 'mem', 15: 'cib', 16: 'caban', 17: 'eznab', 18:
'canac', 19: 'ahau'}
for i in range(n):
```

```
        Haab = input().split()
        Tzolkin = [0, "", 0]
        Haab[0] = int(Haab[0][0:-1])
        Haab[2] = int(Haab[2])

        day_of_the_year = 0
        if Hmonths[Haab[1]] <= 18:
                day_of_the_year = 20 * (Hmonths[Haab[1]] - 1) + Haab[0]
        else:
                day_of_the_year = 360 + Haab[0]
        day_tot = Haab[2] * 365 + day_of_the_year
        Tzolkin[2] = day_tot // 260
        Tzolkin_day = day_tot % 260
        Tzolkin[0] = Tzolkin_day % 13 + 1
        Tzolkin[1] = Tdays[Tzolkin_day % 20]
        print(*Tzolkin, sep = " ")
```

代码运行截图 （至少包含有"Accepted"）

545C. Woodcutters

dp, greedy, 1500, https://codeforces.com/problemset/problem/545/C

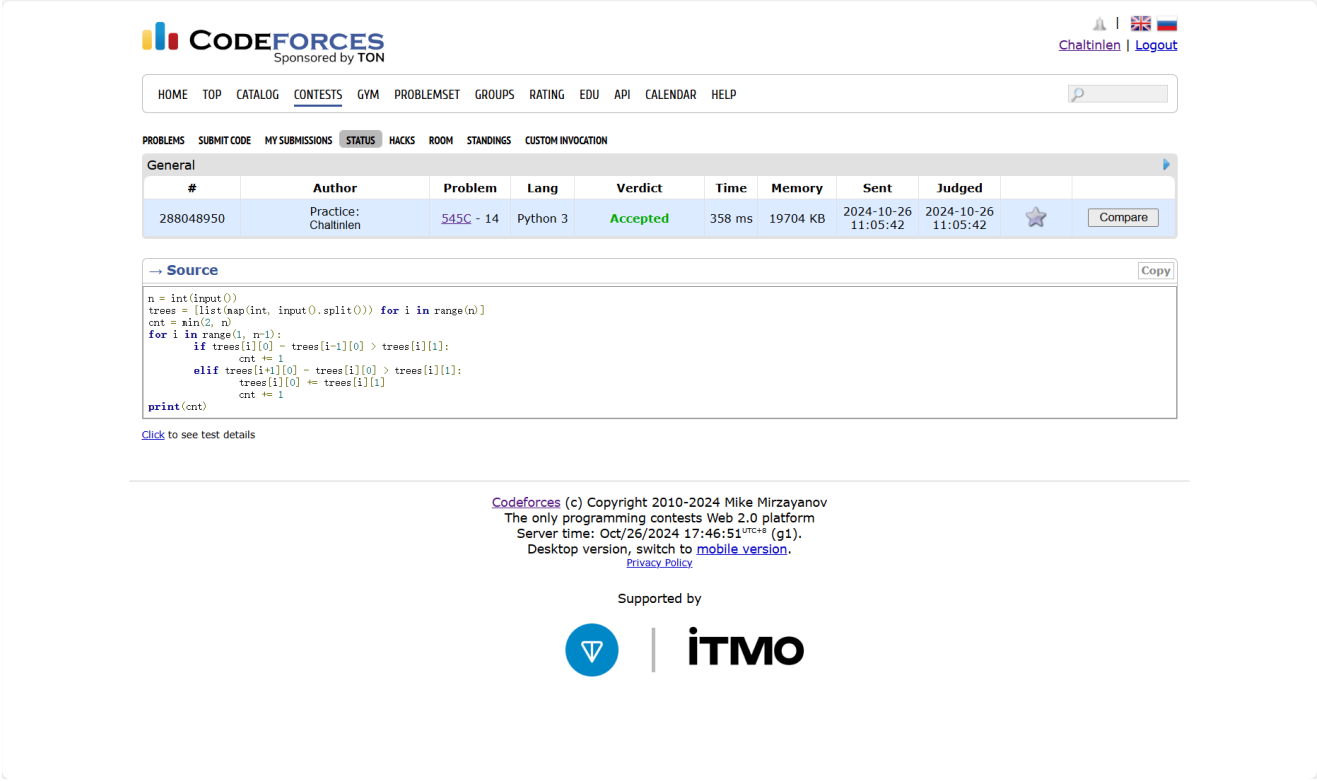
思路：

- 输入保证树的坐标升序排列，无需再排序
- 开头结尾的树一定能砍
- 从前往后遍历每一棵树，如果树往后倒，就把倒下后树梢的位置记为树的新位置

代码：

```
n = int(input())
trees = [list(map(int, input().split())) for i in range(n)]
cnt = min(2, n)
for i in range(1, n-1):
        if trees[i][0] - trees[i-1][0] > trees[i][1]:
                cnt += 1
        elif trees[i+1][0] - trees[i][0] > trees[i][1]:
                trees[i][0] += trees[i][1]
                cnt += 1
print(cnt)
```

代码运行截图 （至少包含有"Accepted"）



01328: Radar Installation

greedy, http://cs101.openjudge.cn/practice/01328/

思路：

- 整体思路类似于OpenJudge - 04100:进程检测
- 注意读取输入时的空行

代码：

```
from math import sqrt, pow
for _ in range(1, int(1e9)):
        n, d = map(int, input().split())
        cnt = 0
        nosol = False
```

```
        if n == 0:
                break
        island = [list(map(int, input().split())) for i in range(n)]
        input()

        xcord = []
        append = xcord.append
        for isl in island:
                if isl[1] > d:
                        print(f"Case {_}: {-1}")
                        nosol = True
                        break
                dist = sqrt(pow(d, 2) - pow(isl[1], 2))
                append((isl[0]-dist, isl[0]+dist))
        if nosol:
                continue

        xcord = sorted(xcord, key = lambda t: t[1])
        rht = -1e9
        for isl in xcord:
                if isl[0] > rht:
                        cnt += 1
                        rht = isl[1]
        print(f"Case {_}: {cnt}")
```

代码运行截图 （至少包含有"Accepted"）

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“计概2024fall每日选做”、CF、LeetCode、洛谷等网站题目。

1. 每日选做有的题目做起来有点痛苦，因为**边界数据**经常与内部数据的状态不同，每次结果差1的时候就特别难受，如OpenJudge - 19757:Saruman's Army，硬控我两天，关键在于写好最后的条件判断

   你的提交记录

#	结果	时间
6	Accepted	2024-10-25
5	Wrong Answer	2024-10-25
4	Wrong Answer	2024-10-25
3	Wrong Answer	2024-10-25
2	Wrong Answer	2024-10-24
1	Wrong Answer	2024-10-24

2. 感觉有一种离了测试数据活不了的感觉，面对莫名其妙的**Wrong Answer**我很难编出一组能找到问题的测试数据，要么去群里找压缩包，群里没有的时候（如上题）不择手段到上网找，如下army.in army.out（要是没有这数据我WA次数还会更多）
   总能给我整出来点莫名其妙的测试数据，比如砍树那个题只有1棵树
3. 幸好月考不计分