

Assignment #C: 五味杂陈

Updated 1148 GMT+8 Dec 10, 2024

2024 fall, Compiled by 陈俊逸、工院

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。
- 2) 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。
- 3) 如果不能在截止前提交作业，请写明原因。

1. 题目

1115. 取石子游戏（10min+思考证明10min）

dfs, <https://www.acwing.com/problem/content/description/1117/>

思路:

哈哈其实这个题的难点在于它给的提示，有了提示以后就是一个easy题目。我下来又想了一下整除大于2是怎么赢的：

赢这个游戏的关键在于掌握主动权！

比如 有a, b两堆石子时，如果我 $a = qb + k$ ，那我可以拿qb，也可以拿qb-b，关键就在于我拿完之后能不能赢，即b, k到底是先拿的赢还是后拿的赢（这就递归了）：

如果b, k时先拿的赢，我第一次就拿qb-b，这样你抓的时候不得不抓b个，这样对于k, b的情况就是我先抓，我赢；

如果b, k后拿的赢，我第一次就拿qb，让你先去拿b, k的情况，这样我就赢了。所以主动权一直都在我手里，除非一开始就是b, k的情况，由于我先拿所以我必输。

如果不给这个提示，可能就要变成tough了吧（反正我是不会往2倍这个方向想，永远不会，，，也可能，说不准）

代码：

```
def dfs(a,b,xuanshou):
    if a<b:
        a,b=b,a
    if a//b>=2 or a%b==0:
        return xuanshou
    else:
        return(dfs(a-b,b,xuanshou+1))
while True:
    a,b=map(int,input().split())
    if a==b==0:
```

```

        break
    cishu=dfs(a,b,0)
    cishu%=2
    if cishu:
        print('lose')
    else:
        print('win')

```

代码运行截图 (至少包含有"Accepted")



25570: 洋葱 (8min)

Matrices, <http://cs101.openjudge.cn/practice/25570>

思路:

一圈一圈绕呗，没啥不好想的，难点在于绕的时候的一些细节。

这道题其实挺感慨的，第一次提交的时候wa了，但是过了一分钟就猜到为题在哪了（n=1的时候）然后debug成功就ac了，感觉debug能力还是和做题数量呈一定程度的正相关的

代码:

```

n=int(input())
ma=[]
for _ in range(n):
    ma.append(list(map(int,input().split())))
ceng=n//2+n%2
ans=0
for c in range(ceng):
    s=ma[c][c]
    x,y=c,c
    while x<n-c-1:
        x+=1
        s+=ma[x][y]
    while y<n-c-1:

```



```

qian[0]=w[0] if w[0]>=0 else 0
ans=1 if w[0]>=0 else 0
for i in range(1,n):
    if w[i]<0:
        fu.append((w[i],i))
        qian[i]=qian[i-1]
    else:
        ans+=1
        qian[i]=qian[i-1]+w[i]
fu.sort(key=lambda x:x[0],reverse=True)
for i in range(len(fu)):
    if qian[fu[i][1]]+fu[i][0]>=0:
        judge=1
        for j in range(fu[i][1],len(qian)):
            qian[j]+=fu[i][0]
            if qian[j]<0:
                judge=0
                for k in range(j,fu[i][1]-1,-1):
                    qian[k]-=fu[i][0]
                break
        ans+=1 if judge==1 else 0
print(ans)

```

代码运行截图 (至少包含有"Accepted")

22067: 快速堆猪 (25min)

辅助栈, <http://cs101.openjudge.cn/practice/22067/>

思路:

翻译一遍题目要求就行了, mini用于记录最小值, judge用于记录最小值的个数, 为0时重新获取最小值

代码:

```

dui=[]
mini=float('inf')
judge=0
while True:
    try:
        shuru=input()
        if shuru[1]=='u':
            s=shuru.split()
            a=int(s[1])
            if mini>a:
                mini=a
                judge=1
            elif mini==a:
                judge+=1
            dui.append(a)
        if shuru[1]=='o' and dui:

```

```

        b=dui.pop()
        if b==mini:
            judge-=1
            if not judge and dui:
                mini=min(dui)
                judge=1
        if shuru[0]=='m' and dui:
            print(mini)
    except EOFError:
        break

```

代码运行截图 (至少包含有"Accepted")


CS101 / 题库 (包括计概、数算题目)

[题目](#)
[排名](#)
[状态](#)
[提问](#)

#47675723提交状态

[查看](#)
[提交](#)
[统计](#)
[提问](#)

状态: **Accepted**

源代码

```

dui=[]
mini=float('inf')
judge=0
while True:
    try:
        shuru=input()
        if shuru[1]=='u':
            s=shuru.split()
            a=int(s[1])
            if mini>a:
                mini=a
                judge=1
            elif mini==a:
                judge+=1
            dui.append(a)
        if shuru[1]=='o' and dui:
            b=dui.pop()

```

基本信息

```

#: 47675723
题目: 22067
提交人: 24n2400011257
内存: 5652kB
时间: 305ms
语言: Python3
提交时间: 2024-12-11 09:07:35

```

20106: 走山路 (1.2h)

Dijkstra, <http://cs101.openjudge.cn/practice/20106/>

思路:

有点像bfs的思路, 从起点开始四个方向遍历, 如果需要的体力小, 就更新newx, newy, 然后把它再添加到queue里, 值得注意的是, 这里的queue是小顶堆, 这是一个优化, 使得每次都去算体力消耗最小的, 那么一旦我遍历到了终点, 一定就是最小的, 这样减少了时间复杂度 (毕竟我没加之前一直tle, 问了ai才反应过来)。还有一点就是dijkstra和bfs的不同我想只是在于添加到queue的方式 (判断条件不太一样), 整体思路都十分相像

代码:

```

import heapq
directions=[[1,0],[0,1],[-1,0],[0,-1]]
m, n, p = map(int, input().split())
ditu = []
for i in range(m):
    ditu.append(list(map(str, input().split())))
for _ in range(p):
    sx, sy, ex, ey = map(int, input().split())

```



```

from collections import deque
di=[(1,0),(0,1),(-1,0),(0,-1)]
T=int(input())
def keyi(nx,ny):
    global n,m,k
    return 0<=nx<n and 0<=ny<m and ((ma[nx][ny]=='#' and (t+1)%k==0) or ma[nx]
[ny]!='#')
for _ in range(T):
    n,m,k=map(int,input().split())
    ma=[]
    for i in range(n):
        shuru=input()
        if 'S' in shuru:
            sx=i
            sy=shuru.index('S')
        if 'E' in shuru:
            ex=i
            ey=shuru.index('E')
    ma.append(shuru)
    queue=deque([(sx,sy,0)])
    visited=set()
    visited.add((sx,sy,0))
    v=[[False]*m for _ in range(n)]
    judge=1
    while queue and judge:
        for _ in range(len(queue)):
            (x,y,t)=queue.popleft()
            if x==ex and y==ey:
                print(t)
                judge=0
                break
            for dx,dy in di:
                nx,ny=x+dx,y+dy
                if keyi(nx,ny) and (nx,ny,t+1) not in queue and not v[nx][ny]:
                    if t%k==0 and (nx,ny,t+1-k) in visited:
                        v[nx][ny]=True
                        continue
                    queue.append((nx,ny,t+1))
                    visited.add((nx,ny,t+1))
    if judge:
        print('Oop!')

```

代码运行截图 (至少包含有"Accepted")

#	结果	时间
18	Accepted	<u>2024-12-13</u>
17	Runtime Error	<u>2024-12-13</u>
16	Runtime Error	<u>2024-12-13</u>
15	Memory Limit Exceeded	<u>2024-12-13</u>
14	Memory Limit Exceeded	<u>2024-12-13</u>
13	Wrong Answer	<u>2024-12-13</u>
12	Time Limit Exceeded	<u>2024-12-13</u>
11	Wrong Answer	<u>2024-12-13</u>
10	Wrong Answer	<u>2024-12-13</u>
9	Time Limit Exceeded	<u>2024-12-13</u>
8	Time Limit Exceeded	<u>2024-12-13</u>
7	Wrong Answer	<u>2024-12-13</u>
6	Wrong Answer	<u>2024-12-13</u>
5	Wrong Answer	<u>2024-12-13</u>
4	<u>Memory Limit</u>	<u>2024-12-13</u>

Exceeded

3	Wrong Answer	2024-12-13
2	Wrong Answer	2024-12-13
1	Wrong Answer	2024-12-13

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。

作业题的收获大多写到思路里了。

每日选做做了几道dp和bfs的题，应为感觉对他俩还不太熟练，但其实最后还是靠室友辅助了一下，不过我对这两个的思路有了更深的理解。

然后就是对计算概论的一些体会：

感觉这学期我已经把它当作一门专业课来学了，花了很多时间，可以说占用了很多本该花在专业课上的时间，然而最后。。。很难评我最后会什么样子。我感觉我室友学c语言的肯定没我花的时间多，到最后题还比我们简单。。。我觉得我们班还有很多同学和我一样，比周围人付出了更多但可能最后结果不尽人意。希望老师能好好考虑一下期末题目的难度，因为我真心觉得和周围人相比的话，不太公平。

QAQ