

第十二次作业

1115. 取石子游戏（在原题目所给的提示下做出来）

```
def dfs(max_rock, min_rock, step):
    max_rock, min_rock = max(max_rock, min_rock), min(max_rock, min_rock)

    # 检查终止条件
    if max_rock // min_rock < 2 and max_rock!=min_rock:
        (一开始没有考虑到两堆石头数目相同的情况)
        return dfs(max_rock - min_rock, min_rock, step + 1)
    else:
        return step

while True:
    a, b = map(int, input().split())
    if a == 0 and b == 0:
        break

    if a // b >= 2 or b // a >= 2:
        print('win')
    else:
        steps = dfs(a, b, 0)
        if steps % 2 == 0:
            print('win')
        else:
            print('lose')
```

代码提交状态: **Accepted**

OJ25570: 洋葱（螺旋矩阵问题）

```
def onion(matrix):
    results=[]
    top,bottom,left,right=0,n-1,0,n-1

    while top<=bottom and left<=right:
        layer=[]
        for col in range(left,right+1):
            layer.append(matrix[top][col])
        top+=1
        for row in range(top,bottom+1):
            layer.append(matrix[row][right])
        right-=1
        if top<=bottom:
            for col in range(right,left-1,-1):
```

```

        layer.append(matrix[bottom][col])
    bottom-=1
    if left<=right:
        for row in range(bottom,top-1,-1):
            layer.append(matrix[row][left])
        left+=1

    results.append(sum(layer))
return max(results)

n=int(input())
m=[list(map(int,input().split()))for _ in range(n)]
print(onion(m))

```

比赛	题目	结果	内存	时间	代码长度	语言	提交时间
 题库 (包括计概、数算题目)	25570: 洋葱	Accepted	3928kB	23ms	754 B	Python3	刚刚

1526C1. Potions(Easy Version)

```

import heapq

n = int(input()) # 输入药水数量
potion = list(map(int, input().split())) # 输入药水的效果值

health = 0 # 初始健康值
count = 0 # 能喝的药水数量
min_heap = [] # 小根堆, 用来存储已饮用的药水效果

for i in range(n):
    health += potion[i] # 喝掉当前的药水
    heapq.heappush(min_heap, potion[i]) # 将药水效果加入堆中
    count += 1 # 增加已喝的药水数量

    # 如果健康值小于 0, 说明健康值不合法, 需要丢掉一个药水
    if health < 0:
        health -= heapq.heappop(min_heap) # 弹出最小的药水, 回退健康值
        count -= 1 # 减少已喝的药水数量

print(count)

```

Contest status							
#	When	Who	Problem	Lang	Verdict	Time	Memory
296045885	Dec/11/2024 15:57UTC	liquoriceliquorice	1526C1 - Potions (Easy Version)	Python 3	Accepted	77 ms	0 KB

OJ22067: 快速堆猪

```

stack=[]
min_stack=[] （一个主栈一个辅助栈）

def push(x):
    stack.append(x)
    if not min_stack or x<= min_stack[-1]:
        min_stack.append(x)

def pop():
    if stack:
        top=stack.pop()
        if top==min_stack[-1]:
            min_stack.pop()

def get_min():
    if min_stack:
        return min_stack[-1]
    return

while True:
    try:
        command=input().strip()
        if command.startswith('push'):
            value=int(command.split()[1])
            push(value)
        elif command.startswith('pop'):
            pop()
        elif command.startswith('min'):
            if get_min() is not None: （这里不能直接写 if get_min()因为如果返回值为 0 的话也会视为 False）
                print(get_min())
    except EOFError:
        break

```

比赛	题目	结果	内存	时间	代码长度	语言	提交时间
 题库（包括计概、数算题目）	22067: 快速堆猪	Accepted	10088kB	416ms	672 B	Python3	刚刚

OJ20106: 走山路

```
import heapq
```

```

m,n,p=map(int,input().split())
mountain = [list(input().split()) for _ in range(m)]

```

```

directions=[(1,0),(-1,0),(0,1),(0,-1)]

def dijkstra(start_x,start_y,end_x,end_y):
    pos=[]
    distance=[[float('inf')]* n for _ in range(m)]
    if mountain[start_x][start_y]=='#':
        return 'NO'
    distance[start_x][start_y]=0
    heapq.heappush(pos,(0,start_x,start_y))

    while pos:
        d,x,y=heapq.heappop(pos)
        if x==end_x and y==end_y:
            return d

        for dx,dy in directions:
            nx=x+dx
            ny=y+dy
            if 0<=nx<m and 0<=ny<n and mountain[nx][ny]!='#':
                if distance[nx][ny]>d+abs(int(mountain[nx][ny])-int(mountain[x][y])):
                    distance[nx][ny]=d+abs(int(mountain[nx][ny])-int(mountain[x][y]))
                    heapq.heappush(pos,(distance[nx][ny],nx,ny))

    return 'NO'

for _ in range(p):
    x1,y1,x2,y2=map(int,input().split())
    print(dijkstra(x1,y1,x2,y2))

```

比赛	题目	结果	内存	时间	代码长度	语言	提交时间
 题库 (包括计概、数算题目)	20106: 走山路	Accepted	3708kB	263ms	967 B	Python3	刚刚

OJ04129: 变换的迷宫

```
from collections import deque
```

```

directions=[(-1,0),(1,0),(0,1),(0,-1)]
def bfs(x,y):
    visited={(0,x,y)}
    q=deque()
    q.append((0,x,y))

    while q:
        time,x,y=q.popleft()

```

```

        for dx,dy in directions:
            nx=x+dx
            ny=y+dy
            ration=(time+1)%K
            if 0<=nx<R and 0<=ny<C and (ration,nx,ny) not in visited:
                spot=maze[nx][ny]
                if spot=='E':
                    return time+1
                elif spot!='#'or ration==0:
                    q.append((time+1,nx,ny))
                    visited.add((ration,nx,ny))

    return 'Oop!'

T=int(input())
for _ in range(T):
    R,C,K=map(int,input().split())
    maze=[list(input()) for _ in range(R)]

    for i in range(R):
        for j in range(C):
            if maze[i][j]=='S':
                print(bfs(i,j))

```

比赛	题目	结果	内存	时间	代码长度	语言	提交时间
 题库（包括计概、数算题目）	04129: 变换的迷宫	Accepted	5116kB	109ms	839 B	Python3	刚刚

感想：巩固了 dfs，螺旋矩阵，heapq，学了辅助栈、dijkstra 的做法。开始看之前的讲义和作业题复习了！