

1. (20min)

1115. 取石子游戏

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

思路：递归，如果最大的大于等于最小的两倍或者与最小的相等，则先手有必胜策略；否则先手只有一种取法，递归到更小情况，类似辗转相除，直到大的是小的数的倍数。

代码：

```
def game(i, x, y):
    if (y//x) >= 2:
        return i
    elif b == a:
        return i
    else:
        return game(1-i, y-x, x)

while True:
    a, b = map(int, input().split())
    if a == b == 0:
        break
    if a > b:
        a, b = b, a
    if game(0, a, b) == 0:
        print('win')
    else:
        print('lose')
```

运行：



The screenshot shows the AcWing website's challenge mode interface. At the top, there's a navigation bar with 'AcWing', '首页', '课程', '题库', and '更多'. A search bar contains '《信息学奥赛一本》'. Below the navigation bar, the '挑战模式' (Challenge Mode) section is active, displaying a Python 3 code editor. The code defines a recursive function 'game(i, x, y)' and a main loop that reads input and calls the function. The code is as follows:

```
1 def game(i, x, y):
2     if (y//x) >= 2:
3         return i
4     elif b == a:
5         return i
6     else:
7         return game(1-i, y-x, x)
8
9
10 while True:
11     a, b = map(int, input().split())
12     if a == b == 0:
13         break
14     if a > b:
15         a, b = b, a
16     if game(0, a, b) == 0:
17         print('win')
18     else:
19         print('lose')
```

Below the code editor, there's a status bar showing '数据有点弱吗？可以申请加强数据' and buttons for '调试代码' (Debug Code) and '提交答案' (Submit Answer). At the bottom, the submission status is 'Accepted'.

2. (20min)

25570: 洋葱

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

思路：基础题，注意边界上的范围即可

代码：

```
n=int(input())
mat=[]
ans=0
for _ in range(n):
    mat.append(list(map(int,input().split())))
dir=[[0,1],[1,0],[0,-1],[-1,0]]
if n%2==1:
    ans=mat[n//2][n//2]
for i in range(n//2):
    num=0
    xi=yi=i
    for j in range(4):
        for k in range(n-1-2*i):
            num+=mat[xi][yi]
            xi+=dir[j][0]
```

```

        yi+=dir[j][1]
    ans=max(ans,num)
print(ans)

```

运行:

#47785851提交状态

查看 提交 统计 提问

状态: Accepted

源代码

```

n=int(input())
mat=[]
ans=0
for _ in range(n):
    mat.append(list(map(int,input().split())))
dir=[[0,1],[1,0],[0,-1],[-1,0]]
if n%2==1:
    ans=mat[n//2][n//2]
for i in range(n//2):
    num=0
    xi=yi=i
    for j in range(4):
        for k in range(n-1-2*i):
            num+=mat[xi][yi]
            xi+=dir[j][0]
            yi+=dir[j][1]
    ans=max(ans,num)
print(ans)

```

基本信息

#: 47785851
 题目: 25570
 提交人: 24n2400011028
 内存: 3996kB
 时间: 34ms
 语言: Python3
 提交时间: 2024-12-17 13:44:05

©2002-2022 POJ 京ICP备20010980号-1

English 帮助 关于

3. (40min)

1526C1. Potions(Easy Version)

greedy, dp, data structures, brute force, *1500,

<https://codeforces.com/problemset/problem/1526/C1>

思路: 贪心, 让前 i 个里喝的药水尽量多并且总和尽量大

代码:

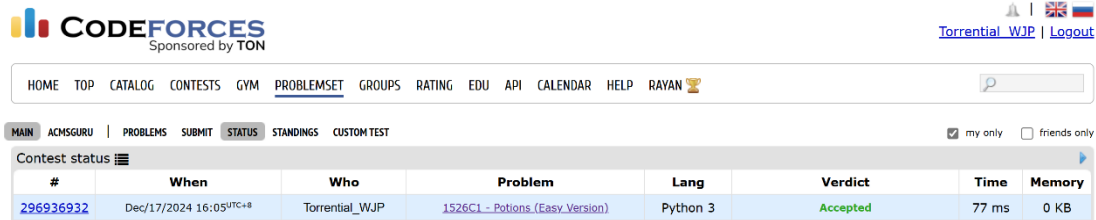
```

import heapq
def drink():
    global potion,n
    num=0
    p=[]
    for k in potion:
        heapq.heappush(p,k)
        num+=k
        if num<0:
            num-=p[0]
            heapq.heappop(p)
    return len(p)

```

```
n=int(input())
potion=list(map(int,input().split()))
print(drink())
```

运行:



CODEFORCES
Sponsored by TON

HOME TOP CATALOG CONTESTS GYM PROBLEMSET GROUPS RATING EDU API CALENDAR HELP RAYAN

MAIN ACMSGURU PROBLEMS SUBMIT STATUS STANDINGS CUSTOM TEST

Contest status

| # | When | Who | Problem | Lang | Verdict | Time | Memory |
|-----------|-------------------------|----------------|---------------------------------|----------|----------|-------|--------|
| 296936932 | Dec/17/2024 16:05 UTC+8 | Torrential_WJP | 1526C1 - Potions (Easy Version) | Python 3 | Accepted | 77 ms | 0 KB |

4. (40min)

22067: 快速堆猪

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

思路: 用辅助栈

代码:

```
a=[]
minstack=[]
while True:
    try:
        s=input().split()
        if s[0]=='min':
            if a:
                print(minstack[-1])
        elif s[0]=='pop':
            if a:
                a.pop()
            if minstack:
                minstack.pop()
        else:
            a.append(int(s[1]))
            if minstack:
                k=minstack[-1]
                minstack.append(min(k,int(s[1])))
            else:
                minstack.append(int(s[1]))
    except EOFError:
        break
```

运行:

#47796621提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: Accepted

源代码

```
a=[]
minstack=[]
while True:
    try:
        s=input().split()
        if s[0]=='min':
            if a:
                print(minstack[-1])
        elif s[0]=='pop':
            if a:
                a.pop()
                if minstack:
                    minstack.pop()
            else:
                a.append(int(s[1]))
                if minstack:
                    k=minstack[-1]
                    minstack.append(min(k,int(s[1])))
                else:
                    minstack.append(int(s[1]))
    except EOFError:
        break
```

基本信息

#: 47796621
题目: 22067
提交人: 24n2400011028
内存: 6032kB
时间: 413ms
语言: Python3
提交时间: 2024-12-17 20:30:38

5. (30min)

20106: 走山路

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

思路: 套 dijkstra 模板即可

代码:

```
import heapq
def dijkstra(x1,y1,x2,y2):
    dir=[[0,1],[0,-1],[-1,0],[1,0]]
    record=[[float('inf')]*n for _ in range(m)]
    record[x1][y1]=0
    q=[]
    heapq.heappush(q,(0,x1,y1))
    while q:
        num,x,y=heapq.heappop(q)
        if x==x2 and y==y2:
            return num
        for dx,dy in dir:
            nx,ny=x+dx,y+dy
            if 0<=nx<m and 0<=ny<n and
mat[nx][ny]!='#':
                if num+abs(int(mat[x][y])-
int(mat[nx][ny]))<record[nx][ny]:
```

```

record[nx][ny]=num+abs(int(mat[x][y])-
int(mat[nx][ny]))

heapq.heappush(q, (record[nx][ny],nx,ny))
    return 'NO'

m,n,p=map(int,input().split())
mat=[]
for _ in range(m):
    mat.append(input().split())
for _ in range(p):
    x1,y1,x2,y2=map(int,input().split())
    if mat[x1][y1]=='#' or mat[x2][y2]=='#':
        print('NO')
    else:
        print(dijkstra(x1,y1,x2,y2))

```

运行:

#47799186提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: **Accepted**

源代码

```

import heapq
def dijkstra(x1,y1,x2,y2):
    dir=[[0,1],[0,-1],[-1,0],[1,0]]
    record=[[float('inf')]*n for _ in range(m)]
    record[x1][y1]=0
    q=[]
    heapq.heappush(q,(0,x1,y1))
    while q:
        num,x,y=heapq.heappop(q)
        if x==x2 and y==y2:
            return num
        for dx,dy in dir:
            nx,ny=x+dx,y+dy
            if 0<=nx<m and 0<=ny<n and mat[nx][ny]!='#':
                if num+abs(int(mat[x][y])-int(mat[nx][ny]))<record[nx][ny]:
                    record[nx][ny]=num+abs(int(mat[x][y])-int(mat[nx][ny]))
                    heapq.heappush(q,(record[nx][ny],nx,ny))
    return 'NO'

m,n,p=map(int,input().split())
mat=[]
for _ in range(m):
    mat.append(input().split())
for _ in range(p):
    x1,y1,x2,y2=map(int,input().split())
    if mat[x1][y1]=='#' or mat[x2][y2]=='#':
        print('NO')

```

基本信息

#: 47799186
 题目: 20106
 提交人: 24n2400011028
 内存: 3716kB
 时间: 276ms
 语言: Python3
 提交时间: 2024-12-17 21:56:36

6. (40min)

04129: 变换的迷宫

bfs, <http://cs101.openjudge.cn/practice/04129/>

思路：套 dfs 模板，唯一的难点是要注意到在 mod k 意义下不同时间到达同一个点可能带来不一样的结果，所以 visited 要用三维数组存储

代码：

```
from collections import deque
def bfs(x1,y1,x2,y2):
    global mat,k,c,r
    dir = [[0, 1], [0, -1], [-1, 0], [1, 0]]
    visited=[[[True]*k for j in range(c)] for p in range(r)]
    visited[x1][y1][0]=False
    q = deque([(0,x1,y1)])
    while q:
        time,x,y=q.popleft()
        if x==x2 and y==y2:
            return time
        for dx,dy in dir:
            nx,ny=x+dx,y+dy
            newtime=time+1
            if 0<=nx<r and 0<=ny<c:
                if newtime%k!=0 and
visited[nx][ny][newtime%k] and mat[nx][ny]!='#':
                    visited[nx][ny][newtime%k]=False
                    q.append((newtime,nx,ny))
                elif newtime%k==0 and
visited[nx][ny][newtime%k]:
                    visited[nx][ny][newtime%k] = False
                    q.append((newtime, nx, ny))
    return "Oop!"

t=int(input())
for _ in range(t):
    r,c,k=map(int,input().split())
    mat=[]
    for a in range(r):
        mat.append(input())
        if 'S' in mat[-1]:
```

```
x1=a
y1=mat[-1].find('S')
if 'E' in mat[-1]:
    x2=a
    y2=mat[-1].find('E')
print(bfs(x1,y1,x2,y2))
```

运行：

#47800173提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: **Accepted**

源代码

```
from collections import deque
def bfs(x1,y1,x2,y2):
    global mat,k,c,r
    dir = [[0, 1], [0, -1], [-1, 0], [1, 0]]
    visited=[[True]*k for j in range(c)] for p in range(r)]
    visited[x1][y1][0]=False
    q = deque([(0,x1,y1)])
    while q:
        time,x,y=q.popleft()
        if x==x2 and y==y2:
            return time
        for dx,dy in dir:
            nx,ny=x+dx,y+dy
            newtime=time+1
            if 0<=nx<r and 0<=ny<c:
                if newtime%k!=0 and visited[nx][ny][newtime%k] and mat[nx][ny]:
                    visited[nx][ny][newtime%k]=False
                    q.append((newtime,nx,ny))
                elif newtime%k==0 and visited[nx][ny][newtime%k]:
                    visited[nx][ny][newtime%k] = False
                    q.append((newtime, nx, ny))
    return "Oop!"
```

基本信息

#: 47800173
题目: 04129
提交人: 24n2400011028
内存: 4784kB
时间: 118ms
语言: Python3
提交时间: 2024-12-17 22:29:05

学习总结和收获：

额外做了一些 greedy 的题目，然而还是感觉考试中如果出现较难的 greedy 就比较看运气了。