Assignment #C: 五味杂陈

Updated 1148 GMT+8 Dec 10, 2024

2024 fall, Complied by 同学的姓名、院系

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

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

思路：

自己做的时候想到了按整除、商为1、商大于1分类，也能AC，但看了答案才意识到商大于1 的时候，该回合取石子的人必胜，dfs可以在这里结束，递归深度更小

代码：

```python
n,m = map(int,input().split())
def dfs(a,b):
    if a%b == 0:
        return True
    if a // b == 1:
        return not dfs(b,a%b)
    return True
while n:
    n,m = max(n,m),min(n,m)
    ans = dfs(n,m)
    print("win" if ans else "lose")
    n,m = map(int,input().split())
```

代码运行截图 （至少包含有"Accepted"）

25570: 洋葱

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

思路：

代码：

```
n = int(input())
matrix = [[] for _ in range(n)]
for i in range(n):
    matrix[i]= list(map(int,input().split()))
m = (n-1)//2 + 1
sum1 = [0] * m
for t in range(m):
    for i in range(t,n-t):
        for j in range(t,n-t):
            sum1[t] += matrix[i][j]
max1 = 0
for i in range(m-1):
    sum1[i] -= sum1[i+1]
print(max(sum1))
```

代码运行截图 （至少包含有"Accepted"）

1526C1. Potions(Easy Version)

greedy, dp, data structures, brute force, *1500, https://codeforces.com/problemset/problem/1526/C1

思路：

代码：

```
n = int(input())
l1 =list(map(int,input().split()))
import heapq
l2 = []
ans = n
life = 0
curr = 0
while curr < n:
    life += l1[curr]
    heapq.heappush(l2,l1[curr])
    if life < 0:
        a = heapq.heappop(l2)
        life -= a
        ans -= 1
    curr += 1
print(ans)
```

代码运行截图 （至少包含有"Accepted"）

22067: 快速堆猪

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

思路：

思维定式了，其实不需要关注每头猪的重量，关注的只有push的猪在当前队列中是否为最小值，如果是，则更新最小值，否则不妨让它为当前最小值

代码：

```
import heapq
stack1 = []
while True:
    try:
        str1 = input()
        if str1[:3] == 'pop':
            if stack1:
                stack1.pop()
        if str1[:3] == 'min' and stack1:
            print(stack1[-1])
        if str1[:4] == 'push':
            a = int(str1[5:])
            if stack1:
                if stack1[-1] > a:
                    stack1.append(a)
                else:
                    stack1.append(stack1[-1])
            else:stack1.append(a)
        # print(stack1)
    except EOFError:break
```

代码运行截图 （至少包含有"Accepted"）

20106: 走山路

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

思路：

感谢老师的dijkstra提示

代码：

```python
import heapq
dx = [1, -1, 0, 0]
dy = [0, 0, 1, -1]
n, m, p = map(int, input().split())
matrix = [list((input().split())) for _ in range(n)]
se = [list(map(int, input().split())) for _ in range(p)]

neighbor = [[] for _ in range(n*m)]
for i in range(n):
    for j in range(m):
        for k in range(4):
            ni = i + dx[k]
            nj = j + dy[k]
            if 0 <= ni < n and 0 <= nj < m:
                if matrix[i][j].isdigit() and matrix[ni][nj].isdigit():
                    neighbor[i*m+j].append((ni*m+nj,
                                           abs(int(matrix[i][j]) - int(matrix[ni][nj]))))
se_ = []
for i in reversed(range(len(se))):
    s = se[i][0]*m + se[i][1]
    e = se[i][2]*m + se[i][3]
    se_.append((s, e))
while se_:
    s, e = se_.pop()
    distance = [float("inf")] * (n*m)
    distance[s] = 0
    vis = set()
    q = [(0, s)]
    while q:
        _,u=heapq.heappop(q)
        if u in vis:
            continue
        vis.add(u)
        for v,w in neighbor[u]:
            # print(distance)
            # print(v,u)
            if distance[v] > distance[u] + w:
                distance[v] = distance[u]+w
                heapq.heappush(q,(distance[v],v))
    print(distance[e] if distance[e] != float("inf") else "NO")
```

代码运行截图 （至少包含有"Accepted"）

04129: 变换的迷宫

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

思路：

代码：

```python
from collections import deque
testnumber = int(input())
dx = [1, -1, 0, 0]
dy = [0, 0, 1, -1]


def check(x, y, t):
    if 0 <= x < R and 0 <= y < C:
        if (t+1) % K == 0:
            return True
        else:
            if matrix[x][y] == '.':
                return True
    return False


for _ in range(testnumber):
    R, C, K = map(int, input().split())
    matrix = [list(input()) for _ in range(R)]
    need_checkstart = True
    need_checkend = True
    for i in range(R):
        if need_checkstart or need_checkend:
            for j in range(C):
                if matrix[i][j] == 'S':
                    sx, sy = i, j
                    matrix[i][j] = '.'
                    need_checkstart = False
                if matrix[i][j] == 'E':
                    ex, ey = i, j
                    matrix[i][j] = '.'
                    need_checkend = False
    stack = deque()
    stack.append((sx, sy, 0))
    inq = set((sx, sy, 0))
    while stack:
        x, y, time = stack.popleft()
        if x == ex and y == ey:
            print(time)
            break
        for i in range(4):
            nx = x+dx[i]
            ny = y+dy[i]
            if check(nx, ny, time):
                if (nx, ny, (1+time) % K) not in inq:
                    stack.append((nx, ny, time+1))
                    inq.add((nx, ny, (1+time) % K))
    else:
        print("Oop!")
```

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。

这些题有的思路很神奇，看答案后学到了很多