

Assignment #D: 十全十美

Updated 1254 GMT+8 Dec 17, 2024

2024 fall, Compiled by <mark>陶嘉瑞-物理学院</mark>

**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

02692: 假币问题

brute force, <http://cs101.openjudge.cn/practice/02692>

思路:
遍历即可

代码:

```
```python
def find_coin(judge):
 for i in range(12):
 for j in range(2):
 coin=[0]*12
 coin[i]=2*j-1
 times=0
 for k in range(3):
 left,right,s=judge[k][0],judge[k][1],judge[k][2]
 sum_left=0
 for l in left:
 sum_left+=coin[ord(l)-65]
 sum_right=0
 for l in right:
```

```

 sum_right+=coin[ord(l)-65]
 if s=='even' and sum_left==sum_right:
 times+=1
 elif s=='up' and sum_left>sum_right:
 times+=1
 elif s=='down' and sum_left<sum_right:
 times+=1
 if times==3:
 return (i,j)
n=int(input())
for i in range(n):
 judge=[]
 for i in range(3):
 judge.append(list(map(str,input().split()))))
 x,y=find_coin(judge)
 a=chr(65+x)
 b='heavy' if y==1 else 'light'
 print('{} is the counterfeit coin and it is {}'.format(a,b))

```

...

代码运行截图 <mark>（至少包含有"Accepted"）</mark>

状态: **Accepted**

源代码

```

def find_coin(judge):
 for i in range(12):
 for j in range(2):
 coin=[0]*12
 coin[i]=2*j-1
 times=0
 for k in range(3):
 left,right,s=judge[k][0],judge[k][1],judge[k][2]

```

基本信息

#: 47789936  
 题目: 02692  
 提交人: talenttao  
 内存: 3644kB  
 时间: 23ms  
 语言: Python3  
 提交时间: 2024-12-17 16:27:39

### 01088: 滑雪

dp, dfs similar, <http://cs101.openjudge.cn/practice/01088>

思路:

从低到高 dp

代码:

```

python
r,c=map(int,input().split())
hill=[]
height=[]
for i in range(r):
 row=list(map(int,input().split()))
 hill.append(row)
 for j in range(c):
 height.append((row[j],i,j))
height.sort()
#print(height)
length=[[-float('inf')]*c for j in range(r)]
h0,x0,y0=height[0]
length[x0][y0]=0
dx=[0,0,1,-1]
dy=[1,-1,0,0]
out=[0]
for i in range(1,r*c):
 h,x,y=height[i]
 length[x][y]=0
 for j in range(4):
 nx,ny=x+dx[j],y+dy[j]
 if 0<=nx<r and 0<=ny<c:
 if length[nx][ny]>=0 and hill[nx][ny]<hill[x][y]:
 length[x][y]=max(length[x][y],length[nx][ny]+1)
 out.append(length[x][y])
print(max(out)+1)

'''

```

代码运行截图 ==（至少包含有"Accepted"）==

状态: [Accepted](#)

源代码

```

r,c=map(int,input().split())
hill=[]
height=[]
for i in range(r):
 row=list(map(int,input().split()))
 hill.append(row)
 for j in range(c):
 height.append((row[j],i,j))
height.sort()
#print(height)
length=[[-float('inf')]*c for j in range(r)]
h0,x0,y0=height[0]

```

基本信息

#: 47791033  
 题目: 01088  
 提交人: talenttao  
 内存: 5144kB  
 时间: 55ms  
 语言: Python3  
 提交时间: 2024-12-17 16:53:27

### 25572: 螃蟹采蘑菇

bfs, dfs, <http://cs101.openjudge.cn/practice/25572/>

思路:

很基础的 bfs

代码:

```
```python
from collections import deque
def bfs(sx1,sy1,dx,dsy):
    queue=deque()
    queue.append((sx1,sy1))
    visited=set()
    visited.add((sx1,sy1))
    while queue:
        x1,y1=queue.popleft()
        if maze[x1][y1]==9 or maze[x1+dx][y1+dsy]==9:
            return 'yes'
        for i in range(4):
            nx1,ny1=x1+dx[i],y1+dy[i]
            if 0<=nx1<n and 0<=ny1<n and 0<=nx1+dx<n and 0<=ny1+dsy<n and
            ((nx1,ny1) not in visited):
                if maze[nx1][ny1]!=1 and maze[nx1+dx][ny1+dsy]!=1:
                    queue.append((nx1,ny1))
                    visited.add((nx1,ny1))
    return 'no'

n=int(input())
maze=[]
start=[]
for i in range(n):
    row=list(map(int,input().split()))
    for j in range(n):
        if row[j]==5:
            start.append((i,j))
    maze.append(row)
#print(start)
sx1,sy1=start[0]
sx2,sy2=start[1]
dx,dsy=sx2-sx1,sy2-sy1
dx=[0,0,1,-1]
dy=[1,-1,0,0]
print(bfs(sx1,sy1,dx,dsy))
```
```

```

代码运行截图 <mark>（至少包含有"Accepted"）</mark>

状态: Accepted

源代码

```
from collections import deque
def bfs(sx1, sy1, dsx, dsy):
    queue = deque()
    queue.append((sx1, sy1))
    visited = set()
    visited.add((sx1, sy1))
    while queue:
        x1, y1 = queue.popleft()
        if maze[x1][y1] == 9 or maze[x1+dsx][y1+dsy] == 9:
```

基本信息

#: 47791867
题目: 25572
提交人: talenttao
内存: 3728kB
时间: 21ms
语言: Python3
提交时间: 2024-12-17 17:19:36

27373: 最大整数

dp, <http://cs101.openjudge.cn/practice/27373/>

思路:

这道题有思路，但是在一些细节的实现上出了点问题导致一直不能 AC，最后参考了答案

代码:

```
```python
def f(string):
 if string == "":
 return 0
 else:
 return int(string)
m = int(input())
n = int(input())
number = input().split()
for i in range(n):
 for j in range(n-1-i):
 if number[j] + number[j+1] > number[j+1] + number[j]:
 number[j], number[j+1] = number[j+1], number[j]
length = []
for num in number:
 length.append(len(num))
dp = [[""] * (m+1) for _ in range(n+1)]
for k in range(m+1):
 dp[0][k] = ""
```

```

for l in range(n+1):
 dp[l][0]="
for i in range(1,n+1):
 for j in range(1,m+1):
 if length[i-1]>j:
 dp[i][j]=dp[i-1][j]
 else:
 dp[i][j]=str(max(f(dp[i-1][j]),int(number[i-1]+dp[i-1][j]-length[i-1])))
print(dp[n][m])

'''

```

代码运行截图 <mark>（至少包含有"Accepted"）</mark>

状态: Accepted

源代码

```

def f(string):
 if string=='':
 return 0
 else:
 return int(string)
m=int(input())
n=int(input())
number=input().split()
for i in range(n):
 for j in range(n-1-i):
 if number[j] + number[j+1] > number[j+1] + number[j]:
 number[j],number[j+1] = number[j+1],number[j]
length=[]

```

基本信息

#: 47793335  
 题目: 27373  
 提交人: talenttao  
 内存: 31372kB  
 时间: 630ms  
 语言: Python3  
 提交时间: 2024-12-17 18:24:17

### ### 02811: 熄灯问题

brute force, <http://cs101.openjudge.cn/practice/02811>

思路:

一开始没想到要穷举第一行, 后面是看了思路后自己写的, 感觉主要就是难想, 实现并不难

代码:

```

```python
import copy
def judge(s):
    room=copy.deepcopy(room0)
    light=[[0]*6 for i in range(5)]
    for i in range(6):
        if s[i]=='1':
            light[0][i]=1

```

```

        for j in range(5):
            nx,ny=dx[j],i+dy[j]
            if 0<=nx<5 and 0<=ny<6:
                room[nx][ny]=abs(1-room[nx][ny])
    for x in range(1,5):
        for y in range(6):
            if room[x-1][y]==1:
                light[x][y]=1
                for j in range(5):
                    nx,ny=x+dx[j],y+dy[j]
                    if 0<=nx<5 and 0<=ny<6:
                        room[nx][ny]=abs(1-room[nx][ny])
    for i in range(6):
        if room[-1][i]==1:
            return (False,light)
    return (True,light)

```

```

dx=[0,0,1,-1,0]
dy=[1,-1,0,0,0]

```

```

first=[]
for i in range(64):
    binn=int(bin(i)[2:])
    first.append(f'{binn:06d}')
room0=[]
for i in range(5):
    room0.append(list(map(int,input().split())))
for i in range(64):
    s=first[i]
    key,light=judge(s)
    if key:
        for j in light:
            print(' '.join(map(str,j)))
        break

```

...

代码运行截图 <mark>（至少包含有"Accepted"）</mark>

状态: Accepted

源代码

```
import copy
def judge(s):
    room=copy.deepcopy(room0)
    light=[[0]*6 for i in range(5)]
    for i in range(6):
        if s[i]=='1':
            light[0][i]=1
            for i in range(5):
```

基本信息

#: 47830416
题目: 02811
提交人: talenttao
内存: 3744kB
时间: 25ms
语言: Python3
提交时间: 2024-12-18 21:27:43

08210: 河中跳房子

binary search, greedy, <http://cs101.openjudge.cn/practice/08210/>

思路:

看到 tag 才想到二分查找, 感觉用对了方法写起来就比较顺利。

代码:

```
```python
def search(s,e,m,rock):
 left,right=s,e
 ans=0
 while left<right:
 mid=(left+right)//2
 s0=0
 num=0
 for i in range(1,n+2):
 if rock[i]-s0<mid:
 num+=1
 else:
 s0=rock[i]
 if num>m:
 right=mid
 else:
 left=mid+1
 ans=mid
 return ans
```

l,n,m=map(int,input().split())

rock=[0]

for i in range(n):

rock.append(int(input()))



```
rock.append(l)
print(search(0,l+1,m,rock))
```

...

状态: Accepted

源代码

```
def search(s,e,m,rock):
 left,right=s,e
 ans=0
 while left<right:
 mid=(left+right)//2
 s0=0
 num=0
 for i in range(1,n+2):
 if rock[i]-s0<mid:
 num+=1
 else:
```

基本信息

#: 47835023  
题目: 08210  
提交人: talenttao  
内存: 5552kB  
时间: 228ms  
语言: Python3  
提交时间: 2024-12-19 11:42:57

代码运行截图 <mark>（至少包含有"Accepted"）</mark>

## ## 2. 学习总结和收获

<mark>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。</mark>

本周持续跟进每日练习，感觉 dfs, bfs 已经掌握的比较好了，但同时感觉像贪心这一类没有“模板”的题目做起来还是有些吃力。像作业中的最后两题都是因为没想对思路而没写出来，但是在看了思路后写起来就没什么困难了。因此在最后的一个星期里要继续练习保持手感。