

Assignment #C: 矩阵、递归、贪心、和dfs similar

Updated 1028 GMT+8 Nov 28, 2023

2023 fall, Compiled by ==苏王捷 工学院==

说明:

本周作业还是难题较多，建议提前开始作业，如果耗时太长，直接找答案看。两个题解，经常更新。所以最好从这个链接下载最新的，<https://github.com/GMyhf/2020fall-cs101>。

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, 学号），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、作业评论有md或者doc。

3) 如果不能在截止前提交作业，请写明原因。

编程环境

==（请改为同学的操作系统、编程环境等）==

操作系统: Windows 11 22H2

Python编程环境: Spyder IDE 5.5.0

1. 题目

如果耗时太长，直接看解题思路，或者源码

CF1881C. Perfect Square

brute force, implementation, 1200, <https://codeforces.com/problemset/problem/1881/C>

黄源森推荐：“一个一般的矩阵”。感觉现在CF problemset第一页的题（难度1000+的）都不是那么好做。

思路：找左上的1/4矩阵，每个对应四个需要相等的元素，找到改变所需的最小步数，加和

代码

```
# # -*- coding: utf-8 -*-
"""
Created on Tue Nov 28 11:09:20 2023

@author: Lenovo
"""

for _ in range(int(input())):
    n=int(input())
    martix=[[0]*(n+2)]
    martix.extend([[0]+list(input())+[0] for i in range(n)])
    martix.extend([[0]*(n+2)])
    ans=0
    for i in range(1,n//2+1):
        for j in range(1,n//2+1):
            a,b,c,d=ord(martix[i][j]),ord(martix[j][n+1-i]),ord(martix[n+1-i]
[n+1-j]),ord(martix[n+1-j][i])
            maxn=max(a,b,c,d)
            ans+=maxn*4-a-b-c-d
    print(ans)
```

代码运行截图 == (至少包含有"Accepted") ==

PROBLEMS SUBMIT CODE MY SUBMISSIONS STATUS HACKS STANDINGS CUSTOM INVOCATION

General										
#	Author	Problem	Lang	Verdict	Time	Memory	Sent	Judged		
234658407	Practice: csxq	1881C - 20	Python 3	Accepted	358 ms	8544 KB	2023-11-28 06:26:24	2023-11-28 06:26:24	★	Compare

→ Source

Copy

```
# -*- coding: utf-8 -*-
"""
Created on Tue Nov 28 11:09:20 2023

@author: Lenovo
"""

for _ in range(int(input())):
    n=int(input())
    martix=[[0]*(n+2)]
    martix.extend([[0]+list(input())+[0] for i in range(n)])
    martix.extend([[0]*(n+2)])
    ans=0
    for i in range(1,n//2+1):
        for j in range(1,n//2+1):
            a,b,c,d=ord(martix[i][j]),ord(martix[j][n+1-i]),ord(martix[n+1-i]
[n+1-j]),ord(martix[n+1-j][i])
            maxn=max(a,b,c,d)
            ans+=maxn*4-a-b-c-d
    print(ans)
```

OJ02694: 波兰表达式

recursion, data structure, <http://cs101.openjudge.cn/practice/02694/>

思路：递归，每出现一个运算符就先进行他的运算

代码

```
#
m=0
def solve():
    global m
    a=l[m]
    m+=1
    if a=='+':
        return solve()+solve()
    elif a=='-':
        return solve()-solve()
    elif a=='*':
        return solve()*solve()
    elif a=='/':
        return solve()/solve()
    else:
        return float(a)

l=[i for i in input().split()]
n=solve()
print("%6f" % n)
```

代码运行截图 == (AC代码截图, 至少包含有"Accepted") ==

#42801838提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: **Accepted**

源代码

```
m=0
def solve():
    global m
    a=l[m]
    m+=1
    if a=='+':
        return solve()+solve()
    elif a=='-':
        return solve()-solve()
    elif a=='*':
        return solve()*solve()
    elif a=='/':
        return solve()/solve()
    else:
        return float(a)

l=[i for i in input().split()]
n=solve()
print("%6f" % n)
```

基本信息

#: 42801838
题目: 02694
提交人: 23n2300011075(才疏学浅)
内存: 3536kB
时间: 27ms
语言: Python3
提交时间: 2023-11-28 11:52:36

OJ18160: 最大连通域面积

dfs similar, <http://cs101.openjudge.cn/practice/18160>

思路: 说是DFS, 其实bfs也很快, 就和Lake Counting一个图。。。直接抄来改了改

代码

```
# # -*- coding: utf-8 -*-
"""
Created on Tue Nov 28 11:37:14 2023

@author: Lenovo
"""

for _ in range(int(input())):
    N,M=map(int,input().split())
    grid=[]
    for _ in range(N):
        row=list(input())
        grid.append(row)
    visited=[[False]*M for _ in range(N)]
    dir_x=[-1,0,1,0,-1,-1,1,1]
    dir_y=[0,-1,0,1,-1,1,-1,1]
    result=[0]
    for i in range(N):
        for j in range(M):
            if grid[i][j]=='w' and not visited[i][j]:
                s=1
                stack=[(i,j)]
                visited[i][j]=True
                while stack:
                    x,y=stack.pop()
                    for k in range(8):
                        nx,ny=x+dir_x[k],y+dir_y[k]
                        if 0<=nx<N and 0<=ny<M and grid[nx][ny]=='w' and not
visited[nx][ny]:
                            stack.append((nx,ny))
                            s+=1
                            visited[nx][ny]=True
                result.append(s)
    print(max(result))
```

代码运行截图 == (AC代码截图, 至少包含有"Accepted") ==

状态: Accepted

源代码

```
# -*- coding: utf-8 -*-
"""
Created on Tue Nov 28 11:37:14 2023

@author: Lenovo
"""

for _ in range(int(input())):
    N,M=map(int,input().split())
    grid=[]
    for _ in range(N):
        row=list(input())
        grid.append(row)
    visited=[[False]*M for _ in range(N)]
    dir_x=[-1,0,1,0,-1,-1,1,1]
    dir_y=[0,-1,0,1,-1,1,-1,1]
    result=[0]
    for i in range(N):
        for j in range(M):
            if grid[i][j]=='W' and not visited[i][j]:
                s=1
                stack=[(i,j)]
                visited[i][j]=True
                while stack:
                    x,y=stack.pop()
                    for k in range(8):
                        nx,ny=x+dir_x[k],y+dir_y[k]
                        if 0<=nx<N and 0<=ny<M and grid[nx][ny]=='W' and not visited[nx][ny]:
                            stack.append((nx,ny))
                            s+=1
                            visited[nx][ny]=True
                    result.append(s)
    print(max(result))
```

基本信息

#: 42801803
题目: 18160
提交人: 23n2300011075(才疏学浅)
内存: 3756kB
时间: 144ms
语言: Python3
提交时间: 2023-11-28 11:49:55

OJ02754: 八皇后

dfs, <http://cs101.openjudge.cn/practice/02754>

思路: DFS 按字典序寻找每一个的次序

代码

```
# # -*- coding: utf-8 -*-
"""
Created on Mon Oct 30 20:48:19 2023

@author: Lenovo
"""

hang=[0]*8
ans=[[0]*8 for _ in range(92)]
num=0
def queen(n):
    global num
    if n==8:
        for i in range(8):
            ans[num][i]=hang[i]+1
        num+=1
```

```

        return
    for i in range(8):
        for j in range(n):
            if i==hang[j] or (j-n)==hang[j]-i or (n-j)==hang[j]-i:
                break
        else:
            hang[n]=i
            queen(n+1)
queen(0)
n=int(input())
for _ in range(n):
    b=int(input())
    for i in range(8):
        print(ans[b-1][i],end=" ")
    print()

```

代码运行截图 == (至少包含有"Accepted") ==

#42801847提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: **Accepted**

源代码

```

# -*- coding: utf-8 -*-
"""
Created on Mon Oct 30 20:48:19 2023

@author: Lenovo
"""

hang=[0]*8
ans=[[0]*8 for _ in range(92)]
num=0
def queen(n):
    global num
    if n==8:
        for i in range(8):
            ans[num][i]=hang[i]+1
        num+=1
        return
    for i in range(8):
        for j in range(n):
            if i==hang[j] or (j-n)==hang[j]-i or (n-j)==hang[j]-i:
                break
        else:
            hang[n]=i
            queen(n+1)
queen(0)
n=int(input())
for _ in range(n):
    b=int(input())
    for i in range(8):
        print(ans[b-1][i],end=" ")
    print()

```

基本信息

#: 42801847
 题目: 02754
 提交人: 23n2300011075(才疏学浅)
 内存: 3620kB
 时间: 45ms
 语言: Python3
 提交时间: 2023-11-28 11:53:39

OJ18146: 乌鸦坐飞机

<http://cs101.openjudge.cn/routine/18146/>

查达闻推荐: 乌鸦坐飞机和装箱子那道题很像, 其实难度不比装箱子高 但是考虑的情况确实不少。

思路: 一开始的时候还少了最后对2的可能分割的讨论, 后已补上;

简单的情况讨论，可以把所有的数分为1,2和4，然后依序排

代码

```
# # -*- coding: utf-8 -*-
"""
Created on Tue Nov 28 10:08:18 2023

@author: Lenovo
"""

n,k=map(int,input().split())
l=list(map(int,input().split()))
count=[0]*5
for i in l:
    count[4]+=i//4
    count[i%4]+=1
if count[3]:
    count[2]+=count[3]
    count[1]+=count[3]
    count[3]=0
if count[4]>n:
    if (count[4]-n)*2+count[2]+count[1]<=n*2:
        print("YES")
    else:
        print("NO")
else:
    left=n-count[4]
    can=min(left,count[1],count[2])
    count[2]-=can
    count[1]-=can
    left-=can
    if not count[2]:
        if count[1]<=(left+n)*2:
            print("YES")
        else:
            print("NO")
    elif not count[1]:
        if count[2]<=int(left*1.5)+n*2:
            print("YES")
        else:
            print("NO")
    else:
        if count[2]+count[1]<=n*2:
            print("YES")
        else:
            print("NO")
```

代码运行截图 == (AC代码截图，至少包含有"Accepted") ==

状态: **Accepted**

源代码

```
# -*- coding: utf-8 -*-
"""
Created on Tue Nov 28 10:08:18 2023

@author: Lenovo
"""

n,k=map(int,input().split())
l=list(map(int,input().split()))
count=[0]*5
for i in l:
    count[4]+=i//4
    count[i%4]+=1
if count[3]:
    count[2]+=count[3]
    count[1]+=count[3]
    count[3]=0
if count[4]>n:
    if (count[4]-n)*2+count[2]+count[1]<=n*2:
        print("YES")
    else:
        print("NO")
else:
    left=n-count[4]
    can=min(left,count[1],count[2])
    count[2]-=can
    count[1]-=can
    left-=can
    if not count[2]:
        if count[1]<=(left+n)*2:
            print("YES")
        else:
            print("NO")
    elif not count[1]:
        if count[2]<=int(left*1.5)+n*2:
            print("YES")
        else:
            print("NO")
    else:
        if count[2]+count[1]<=n*2:
            print("YES")
        else:
            print("NO")
```

基本信息

#: 42801606
题目: 18146
提交人: 23n2300011075(才疏学浅)
内存: 3704kB
时间: 30ms
语言: Python3
提交时间: 2023-11-28 11:32:45

OJ02287: 田忌赛马

greedy, <http://cs101.openjudge.cn/practice/02287>

思路: 老思路了

代码

```
# def check(a,b):
    ans=0
    l1,r1=0,len(a)-1
    l2,r2=0,len(b)-1
    while l1<=r1 and l2<=r2:
        if b[r2]>a[r1]:
            r2-=1
            r1-=1
            ans+=200
        elif b[l2]>a[l1]:
            l1+=1
            l2+=1
```



```

        ans+=200
    elif b[l2]==a[r1]:
        l2+=1
        r1-=1
        ans+=0
    else:
        r1-=1
        l2+=1
        ans-=200
    return ans

while True:
    n=int(input())
    if n==0:
        break
    tian=list(map(int,input().split()))
    king=list(map(int,input().split()))
    tian.sort()
    king.sort()
    maxi=check(king,tian)
    print(maxi)

```

代码运行截图 == (AC代码截图, 至少包含有"Accepted") ==

#42801866提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: **Accepted**

源代码

```

def check(a,b):
    ans=0
    l1,r1=0,len(a)-1
    l2,r2=0,len(b)-1
    while l1<=r1 and l2<=r2:
        if b[r2]>a[r1]:
            r2-=1
            r1-=1
            ans+=200
        elif b[l2]>a[l1]:
            l1+=1
            l2+=1
            ans+=200
        elif b[l2]==a[r1]:
            l2+=1
            r1-=1
            ans+=0
        else:
            r1-=1
            l2+=1
            ans-=200
    return ans

while True:
    n=int(input())
    if n==0:
        break
    tian=list(map(int,input().split()))
    king=list(map(int,input().split()))
    tian.sort()
    king.sort()
    maxi=check(king,tian)
    print(maxi)

```

基本信息

#: 42801866
 题目: 02287
 提交人: 23n2300011075(才疏学浅)
 内存: 3820kB
 时间: 57ms
 语言: Python3
 提交时间: 2023-11-28 11:55:08

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“每日选做”中每天推出的2题目、CF、LeetCode、洛谷等网站题目。==

忽感每日选做与期末题一样了，干脆把上学期的期末题全做了一遍，两个小时堪堪AC，如果最后一题一开始提示用BFS的话，就不会浪费那么多时间在DFS算法上了：（