# Assignment #A: dp & bfs

Updated 2 GMT+8 Nov 25, 2024

2024 fall, Complied by <mark>同学的姓名、院系</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

### LuoguP1255 数楼梯

dp, bfs, https://www.luogu.com.cn/problem/P1255

思路：每步可以走一步上来也可以走两步，所以是前面两项相加

代码：

```python

N=int(input())

dp=[0]\*(N+1)

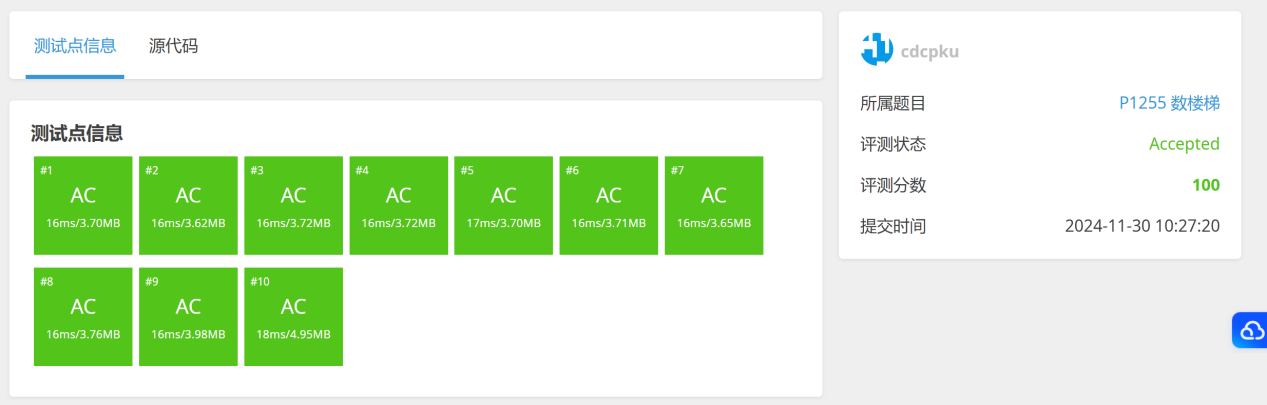
dp[0]=dp[1]=1

for i in range(2,N+1):

dp[i]=dp[i-1]+dp[i-2]

print(dp[N])

```

代码运行截图 <mark>（至少包含有"Accepted"）</mark>

### 27528: 跳台阶

dp, http://cs101.openjudge.cn/practice/27528/

思路：每一步的情况数是前面使用情况数总和+1，用dp1储存每一步及之前的步数总和

代码：

```python

N=int(input())

dp=[0]\*(N+1)

dp1=[0]\*(N+1)

dp[1]=1

dp1[1]=1

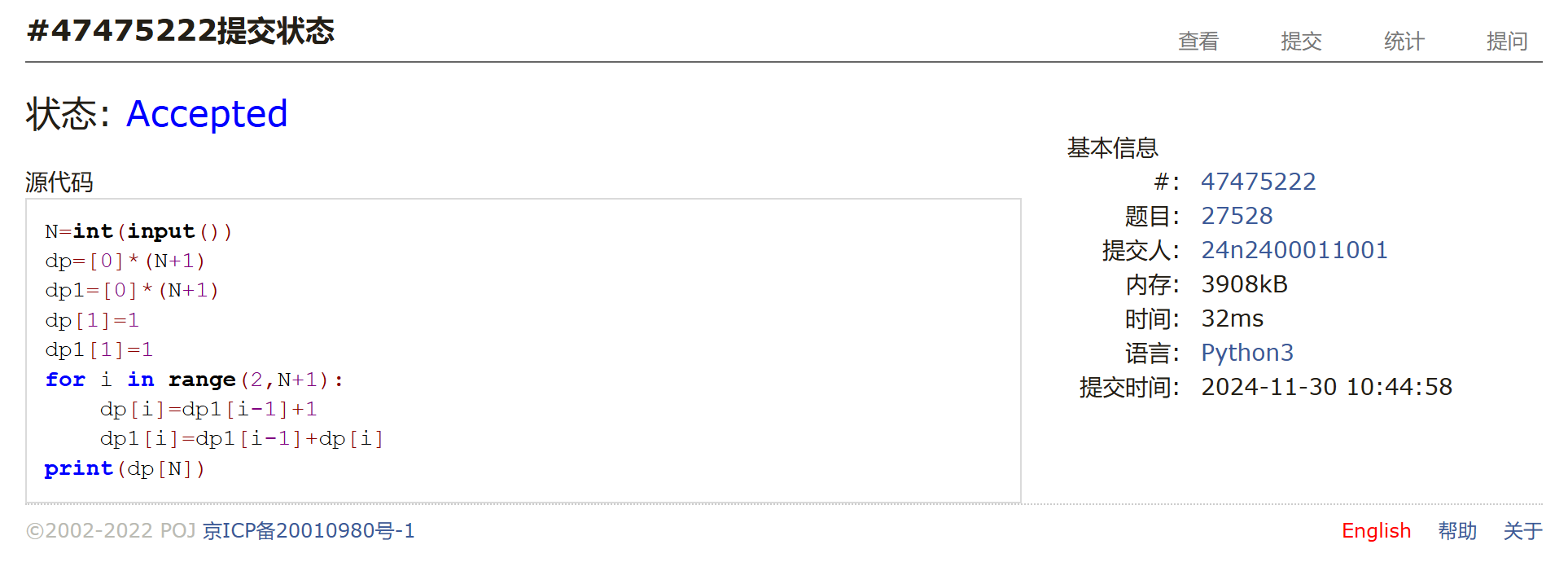
for i in range(2,N+1):

dp[i]=dp1[i-1]+1

dp1[i]=dp1[i-1]+dp[i]

print(dp[N])

```

代码运行截图 ==（至少包含有"Accepted"）==

### 474D. Flowers

dp, https://codeforces.com/problemset/problem/474/D

思路：i>=k第i个可以是红，此时情况数与i-1时相同：也可以是白花，这样的话前面k-1个也要是白花，情况数就与i-k时相同，自己做时死活没想到这点，看了题解恍然大悟

代码：

```python

t,k=map(int,input().split())

dp=[0]\*(10\*\*5+1)

for i in range(10\*\*5+1):

if i<k:

dp[i]=1

else:

dp[i]=(dp[i-1]+dp[i-k])%(10\*\*9+7)

dp1=[0]\*(10\*\*5+1)

dp1[0]=1

for i in range(1,10\*\*5+1):

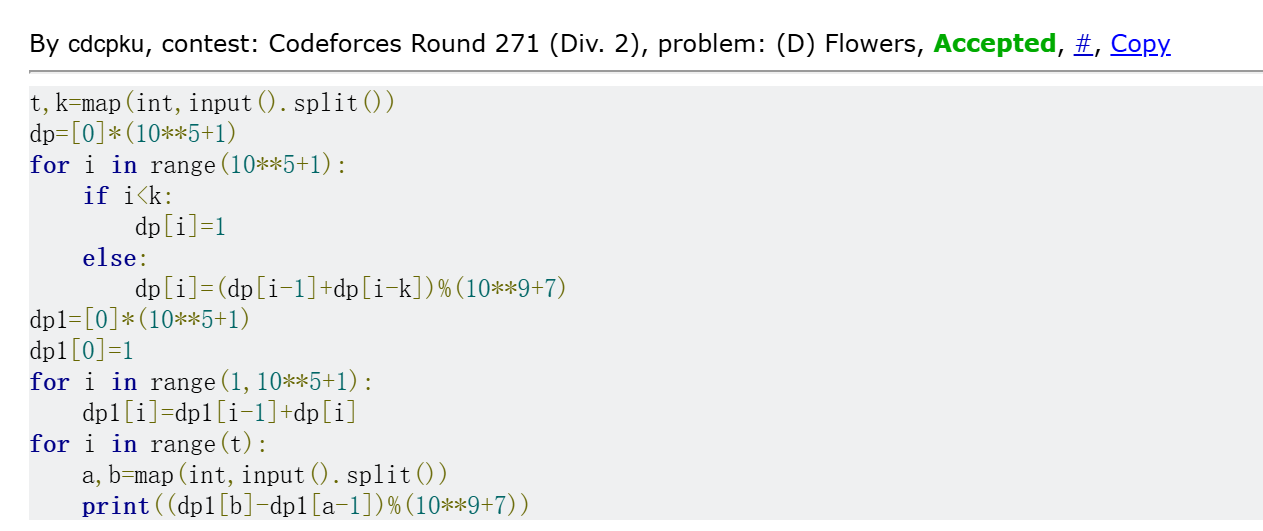
dp1[i]=dp1[i-1]+dp[i]

for i in range(t):

a,b=map(int,input().split())

print((dp1[b]-dp1[a-1])%(10\*\*9+7))

```

代码运行截图 <mark>（至少包含有"Accepted"）</mark>

### LeetCode5.最长回文子串

dp, two pointers, string, https://leetcode.cn/problems/longest-palindromic-

substring/

思路：用dp[i][j]表示从s[i]到s[j]是否为回文字符串

代码：

```python

class Solution:

def longestPalindrome(self, s: str) -> str:

def huiwen(s):

if len(s)==1:

return True

if len(s)<1:

return False

a=0

b=len(s)-1

while a<b:

if s[a]==s[b]:

a+=1

b-=1

else:

return False

return True

dp=[[False]\*len(s) for i in range(len(s))]

for i in range(len(s)-1,-1,-1):

for j in range(i,len(s)):

if i+1<=j-1:

if dp[i+1][j-1]:

dp[i][j]=True if s[i]==s[j] else False

else:

dp[i][j]=huiwen(s[i:j+1])

MAX=0

output=[]

for i in range(len(s)):

for j in range(i,len(s)):

if dp[i][j]:

if j-i+1>MAX:

MAX=j-i+1

output=s[i:j+1]

return ''.join(output)

```

代码运行截图 <mark>（至少包含有"Accepted"）</mark>

### 12029: 水淹七军

bfs, dfs, http://cs101.openjudge.cn/practice/12029/

思路：一开始加保护圈做的，尽管用sys一次性输入了仍因未知原因不断re，去了保护圈一遍过了，之后瞪了半天才发现之前保护圈加少了。。。

代码：

```python

import sys

sys.setrecursionlimit(10\*\*6)

def dfs(C,x,y):

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

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

for i in range(4):

nx,ny=x+dx[i],y+dy[i]

if 0<=nx<M and 0<=ny<N and C[nx][ny][0]<C[x][y][0]:

C[nx][ny][0]=C[x][y][0]

C[nx][ny][1]=True

dfs(C,nx,ny)

data=sys.stdin.read().split()

idx=0

k=int(data[idx])

idx+=1

result=[]

for i in range(k):

M,N=map(int,data[idx:idx+2])

idx+=2

C=[]

for j in range(M):

a=list(map(int,data[idx:idx+N]))

idx+=N

b=[]

for l in a:

b.append([l,False])

C.append(b)

I,J=map(int,data[idx:idx+2])

idx+=2

P=int(data[idx])

idx+=1

for j in range(P):

X,Y=map(int,data[idx:idx+2])

idx+=2

dfs(C,X-1,Y-1)

result.append('Yes' if C[I-1][J-1][1] else 'No')

sys.stdout.write('\n'.join(result)+'\n')

```

代码运行截图 <mark>（至少包含有"Accepted"）</mark>

### 02802: 小游戏

bfs, http://cs101.openjudge.cn/practice/02802/

思路：思路还算好想，对模板熟了之后bfs的代码还算好写，但是一直presentation error，修了一天多也不知道咋回事，最后才发现输出结果的冒号后面有个空格。。。

代码：

```python

from collections import deque

def bfs(matrix,x1,y1,x2,y2):

MIN=float('inf')

q=deque()

inq=set()

q.append((x1,y1,0,0,0))

inq.add((x1,y1))

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

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

while q:

x,y,step,deltax,deltay=q.popleft()

for i in range(4):

nx,ny=x+dx[i],y+dy[i]

if 0<=nx<=w+1 and 0<=ny<=h+1:

if nx==x2 and ny==y2:

if dx[i]==deltax and dy[i]==deltay:

MIN=min(MIN,step)

break

else:

MIN=min(MIN,step+1)

break

if matrix[ny][nx]!='X' and (nx,ny) not in inq:

if dx[i]==deltax and dy[i]==deltay:

q.append((nx,ny,step,dx[i],dy[i]))

inq.add((nx,ny))

else:

q.append((nx,ny,step+1,dx[i],dy[i]))

inq.add((nx,ny))

return MIN

n=1

while True:

w,h=map(int,input().split())

if w==h==0:

break

print(f'Board #{n}:')

matrix=[]

matrix.append([' ']\*(w+2))

for i in range(h):

matrix.append([' ']+list(input())+[' '])

matrix.append([' ']\*(w+2))

m=0

while True:

m+=1

x1,y1,x2,y2=map(int,input().split())

if x1==x2==y1==y2==0:

break

if bfs(matrix,x1,y1,x2,y2)==float('inf'):

print(f'Pair {m}: impossible.')

else:

print(f'Pair {m}: {bfs(matrix,x1,y1,x2,y2)} segments.')

print()

n+=1

```

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>

这周作业感觉难度差距过大，前两题非常简单，第三题苦思冥想不得其解（不过复习上节课pdf的时候发现跟核电站那题基本一个思路。。。还是不熟），第四题用二维dp的思路也是不看题解想不出，后两题思路上反而简单了但格式要求真是恶心人，希望考试时别有这么诡异的要求