

Assignment #10: dp & bfs

Updated 2 GMT+8 Nov 25, 2024

2024 fall, Compiled by 洪千濠 工学院

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：耗时30min

代码：

```
def count_ways(n):
    if n == 1:
        return 1
    elif n == 2:
        return 2

    a, b = 1, 2
    for i in range(3, n + 1):
        a, b = b, a + b
    return b

def main():
    N = int(input())

    result = count_ways(N)
    print(result)

if __name__ == "__main__":
    main()
```

代码运行截图 (至少包含有"Accepted")

测试点信息

#1 AC 16ms/3.70MB	#2 AC 17ms/3.62MB	#3 AC 16ms/3.63MB	#4 AC 16ms/3.63MB	#5 AC 17ms/3.63MB	#6 AC 16ms/3.67MB	#7 AC 16ms/3.67MB
#8 AC 16ms/3.69MB	#9 AC 16ms/3.67MB	#10 AC 16ms/3.74MB				

评测状态

评测分数

提交时间

27528: 跳台阶

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

思路：耗时30min

代码：

```
def count_ways(N):
    if N == 0:
        return 0
    if N == 1:
        return 1

    return 2 ** (N - 1)

def main():
    N = int(input())

    result = count_ways(N)
    print(result)

if __name__ == "__main__":
    main()
```

代码运行截图 == (至少包含有"Accepted") ==

状态: Accepted

源代码

```
def count_ways(N):
    if N == 0:
        return 0
    if N == 1:
        return 1

    return 2 ** (N - 1)

def main():
    N = int(input())

    result = count_ways(N)
```

474D. Flowers

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

思路: 耗时2h

代码:

```
t,k=map(int,input().split())
dp=list([0]*2 for _ in range(100001))
p=[0]*(100001)
dp[0][0]=1
dp[0][1]=0
p[0]=1
for i in range(1,100001):
    dp[i][0]=(dp[i-1][0]+dp[i-1][1])%1000000007
    if i>=k:
        dp[i][1]=(dp[i-k][0]+dp[i-k][1])%1000000007
    p[i]=(p[i-1]+dp[i][0]+dp[i][1])%1000000007
for _ in range(t):
    a,b=map(int,input().split())
    print((p[b]-p[a-1])%1000000007)
```

代码运行截图 (至少包含有"Accepted")

293437020	Nov/27/2024 12:24 UTC+8	Slivahong01	474D - Flowers	Python 3	Accepted	749 ms	20100 KB
-----------	-------------------------	-------------	----------------	----------	----------	--------	----------

LeetCode5.最长回文子串

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

思路：耗时50min,高度还原leetcode官方题解，理解花了好久

代码：

```
class Solution:
    def expandAroundCenter(self, s, left, right):
        while left >= 0 and right < len(s) and s[left] == s[right]:
            left -= 1
            right += 1
        return left + 1, right - 1

    def longestPalindrome(self, s: str) -> str:
        start, end = 0, 0
        for i in range(len(s)):
            left1, right1 = self.expandAroundCenter(s, i, i)
            left2, right2 = self.expandAroundCenter(s, i, i + 1)
            if right1 - left1 > end - start:
                start, end = left1, right1
            if right2 - left2 > end - start:
                start, end = left2, right2
        return s[start: end + 1]
```

代码运行截图 (至少包含有"Accepted")

通过

 Sleepy GoodallApg 提交于 2024.11.27 10:26



「感恩季」限时福利！
买 1 得 2 同时可享力扣 365 天 Plus 会员

🕒 执行用时分布



339 ms | 击败 68.30% 🍏

💡 复杂度分析

15.0%

12029: 水淹七军

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

思路：耗时2h

代码：

```
from collections import deque
import sys
input = sys.stdin.read

def is_valid(x, y, m, n):
    return 0 <= x < m and 0 <= y < n

def bfs(start_x, start_y, start_height, m, n, h, water_height):
    dx = [-1, 1, 0, 0]
    dy = [0, 0, -1, 1]
    q = deque([(start_x, start_y, start_height)])
    water_height[start_x][start_y] = start_height

    while q:
        x, y, height = q.popleft()
        for i in range(4):
            nx, ny = x + dx[i], y + dy[i]
            if is_valid(nx, ny, m, n) and h[nx][ny] < height:
```

```

        if water_height[nx][ny] < height:
            water_height[nx][ny] = height
            q.append((nx, ny, height))

def main():
    data = input().split()
    idx = 0
    k = int(data[idx])
    idx += 1
    results = []

    for _ in range(k):
        m, n = map(int, data[idx:idx + 2])
        idx += 2
        h = []
        for i in range(m):
            h.append(list(map(int, data[idx:idx + n])))
            idx += n
        water_height = [[0] * n for _ in range(m)]

        i, j = map(int, data[idx:idx + 2])
        idx += 2
        i, j = i - 1, j - 1

        p = int(data[idx])
        idx += 1

        for _ in range(p):
            x, y = map(int, data[idx:idx + 2])
            idx += 2
            x, y = x - 1, y - 1
            if h[x][y] <= h[i][j]:
                continue
            bfs(x, y, h[x][y], m, n, h, water_height)

        results.append("Yes" if water_height[i][j] > 0 else "No")

    sys.stdout.write("\n".join(results) + "\n")

if __name__ == "__main__":
    main()

```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

源代码

```

from collections import deque

```

02802: 小游戏

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

思路：耗时40min

代码：

```
import heapq
num1=1
while True:
    w,h=map(int,input().split())
    if w==0 and h==0:
        break
    print(f"Board #{num1}:")
    martix=[[" "]*(w+2)]+[[[" "+list(input())+" " for _ in range(h)]+[[" "]*(w+2)]]
    dir=[(0,1),(0,-1),(1,0),(-1,0)]
    num2=1
    while True:
        x1,y1,x2,y2=map(int,input().split())
        if x1==0 and x2==0 and y1==0 and y2==0:
            break
        queue,flag=[],False
        vis=set()
        heapq.heappush(queue,(0,x1,y1,-1))
        martix[y2][x2]=" "
        vis.add((-1,x1,y1))
        while queue:
            step,x,y,dirs=heapq.heappop(queue)
            if x==x2 and y==y2:
                flag=True
                break
            for i,(dx,dy) in enumerate(dir):
                px,py=x+dx,y+dy
                if 0<=px<=w+1 and 0<=py<=h+1 and (i,px,py) not in vis and martix[py][px]!="X":
                    vis.add((i,px,py))
                    heapq.heappush(queue,(step+(dirs!=i),px,py,i))
        if flag:
            print(f"Pair {num2}: {step} segments.")
        else:
            print(f"Pair {num2}: impossible.")
        martix[y2][x2]="X"
        num2+=1
    print()
    num1+=1
```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

源代码

```
import heapq
num1=1
while True:
    w,h=map(int,input().split())
    if w==0 and h==0:
```

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。

感觉越来越好了，这次对题解的依赖降低了，但优化依然有些想不到，继续学习中