

Assignment #10: dp & bfs

Updated 2 GMT+8 Nov 25, 2024

2024 fall, Compiled by 徐贤天, 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:

代码:

```
n = int(input())
dp = [0] * (n+1)
dp[0] = 1
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 (至少包含有"Accepted")

R191226181 记录详情

编程语言	代码长度	用时	内存
Python 3	125B	161ms	4.93MB

测试点信息 源代码

源代码 [复制](#)

```
n = int(input())
dp = [0] * (n+1)
dp[0] = 1
dp[1] = 1
for i in range(2, n+1):
    dp[i] = dp[i-1] + dp[i-2]
print(dp[n])
```



所属题目	P1255 数楼梯
评测状态	Accepted
评测分数	100
提交时间	2024-11-26 15:42:22

27528: 跳台阶

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

思路:

代码:

```
print(2**(int(input())-1))
```

代码运行截图 == (至少包含有"Accepted") ==

#47408505提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: Accepted

源代码

```
print(2**(int(input())-1))
```

基本信息

#:	47408505
题目:	27528
提交人:	24n2400011033
内存:	3604kB
时间:	30ms
语言:	Python3
提交时间:	2024-11-26 16:11:28

©2002-2022 POJ 京ICP备20010980号-1

[English](#) [帮助](#) [关于](#)

474D. Flowers

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

思路:

代码:

```

t, k = map(int, input().split())
ans = []
dp = [1] * 100001
for i in range(k, 100001):
    dp[i] = (dp[i-1] + dp[i-k]) % 1000000007
the_sum = [0] * 100001
for i in range(1, 100001):
    the_sum[i] = (the_sum[i-1] + dp[i]) % 1000000007
for _ in range(t):
    l, r = map(int, input().split())
    ans.append((the_sum[r] - the_sum[l-1]) % 1000000007)
print(*ans, sep = '\n')

```

代码运行截图 (至少包含有"Accepted")

By Aunixt, contest: Codeforces Round 271 (Div. 2), problem: (D) Flowers, **Accepted**, #, [Copy](#)

```

t, k = map(int, input().split())
ans = []
dp = [1] * 100001
for i in range(k, 100001):
    dp[i] = (dp[i-1] + dp[i-k]) % 1000000007
the_sum = [0] * 100001
for i in range(1, 100001):
    the_sum[i] = (the_sum[i-1] + dp[i]) % 1000000007
for _ in range(t):
    l, r = map(int, input().split())
    ans.append((the_sum[r] - the_sum[l-1]) % 1000000007)
print(*ans, sep = '\n')

```

→Judgement Protocol

Test: #1, time: 93 ms., memory: 7088 KB, exit code: 0, checker exit code: 0, verdict: OK

Input

3 2
1 3
2 3
4 4

Output

6
5
5

Answer

6
5
5

Checker Log

ok 3 number(s): "6 5 5"

Test: #2, time: 62 ms., memory: 3740 KB, exit code: 0, checker exit code: 0, verdict: OK

Input

139 99252
70888 79845
2746 73795
86884 83479
28655 47059
7868 28059
87506 97173
61498 61785
16888 85881

LeetCode5.最长回文子串

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

思路:

代码:

```

s = input()
n = len(s)
start, end = 0, 0
for i in range(n):
    l1, r1 = i, i
    while l1 >= 0 and r1 <= n-1 and s[l1] == s[r1]:

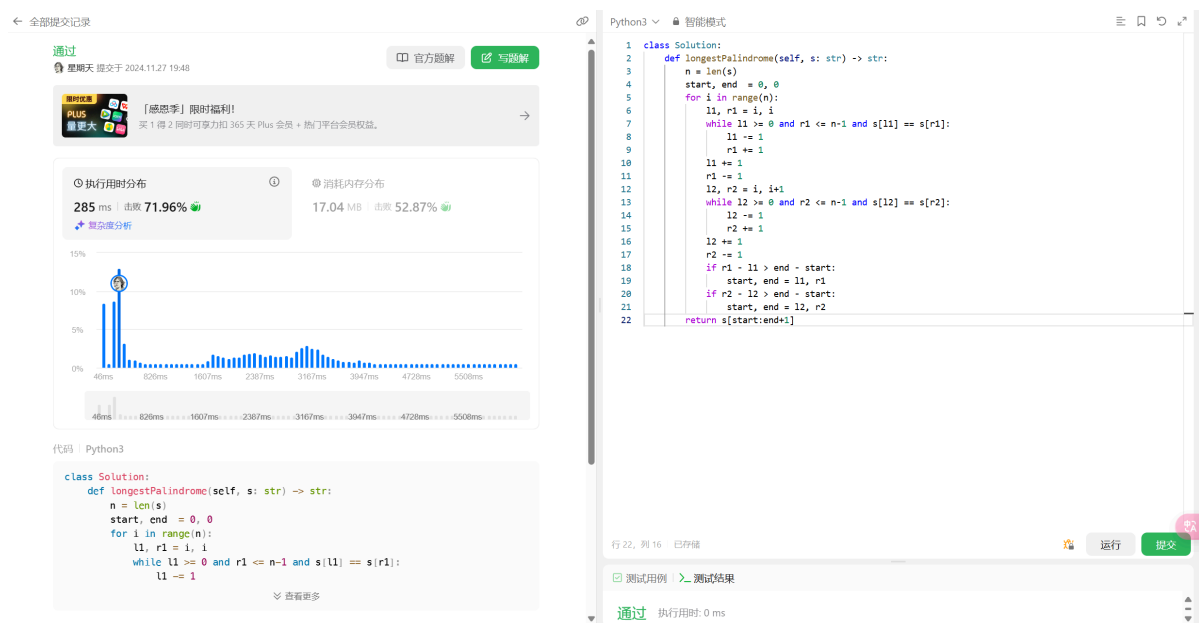
```

```

        l1 -= 1
        r1 += 1
    l1 += 1
    r1 -= 1
    l2, r2 = i, i+1
    while l2 >= 0 and r2 <= n-1 and s[l2] == s[r2]:
        l2 -= 1
        r2 += 1
    l2 += 1
    r2 -= 1
    if r1 - l1 > end - start:
        start, end = l1, r1
    if r2 - l2 > end - start:
        start, end = l2, r2
print(s[start:end+1])

```

代码运行截图 (至少包含有"Accepted")



12029: 水淹七军

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

思路:

代码:

```

import sys
sys.setrecursionlimit(300000)

directions = [(1, 0), (-1, 0), (0, 1), (0, -1)]

def dfs(x, y, height, water_height):

```

```

    for dx, dy in directions:
        nx, ny = x + dx, y + dy
        if 0 <= nx < m and 0 <= ny < n and matrix[nx][ny] < height:
            if water_height[nx][ny] < height:
                water_height[nx][ny] = height
                dfs(nx, ny, height, water_height)

ans = []
data = sys.stdin.read().split()
idx = 0
k = int(data[idx])
idx += 1
for _ in range(k):
    m, n = map(int, data[idx:idx + 2])
    idx += 2
    matrix = []
    for _ in range(m):
        matrix.append(list(map(int, data[idx:idx + n])))
        idx += n
    i, j = map(int, data[idx:idx + 2])
    idx += 2
    i, j = i - 1, j - 1
    p = int(data[idx])
    idx += 1
    water_height = [[0] * n for _ in range(m)]
    for _ in range(p):
        x, y = map(int, data[idx:idx + 2])
        idx += 2
        x, y = x - 1, y - 1
        if matrix[x][y] < matrix[i][j]:
            continue
        dfs(x, y, matrix[x][y], water_height)
    ans.append('Yes' if water_height[i][j] > 0 else 'No')
sys.stdout.write('\n'.join(ans) + '\n')

```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

源代码

```
import sys
sys.setrecursionlimit(300000)

directions = [(1, 0), (-1, 0), (0, 1), (0, -1)]

def dfs(x, y, height, water_height):
    for dx, dy in directions:
        nx, ny = x + dx, y + dy
        if 0 <= nx < m and 0 <= ny < n and matrix[nx][ny] < height:
            if water_height[nx][ny] < height:
                water_height[nx][ny] = height
                dfs(nx, ny, height, water_height)

ans = []
data = sys.stdin.read().split()
idx = 0
k = int(data[idx])
idx += 1
for _ in range(k):
    m, n = map(int, data[idx:idx + 2])
    idx += 2
    matrix = []
    for _ in range(m):
        matrix.append(list(map(int, data[idx:idx + n])))
        idx += n
    i, j = map(int, data[idx:idx + 2])
    idx += 2
    i, j = i - 1, j - 1
    p = int(data[idx])
    idx += 1
    water_height = [[0] * n for _ in range(m)]
    for _ in range(p):
        x, y = map(int, data[idx:idx + 2])
        idx += 2
        x, y = x - 1, y - 1
        if matrix[x][y] < matrix[i][j]:
            continue
```

基本信息

#: 47445227
题目: 12029
提交人: 24n2400011033
内存: 13180kB
时间: 219ms
语言: Python3
提交时间: 2024-11-28 15:45:08

02802: 小游戏

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

思路:

把segment和idx数据加入数组中, 每次搜索如果不等于idx, 则说明转向了, segment需要加1

代码:

```
#pylint:skip-file
from collections import deque
dx = [1, -1, 0, 0]
dy = [0, 0, 1, -1]
def bfs(x1, y1, x2, y2):
    global ans
    q = deque([(y1, x1, 0, -1)])
    inq = {(y1, x1)}
    while q:
        y, x, segment, idx = q.popleft()
        if y == y2 and x == x2:
            ans = min(ans, segment)
            continue
        for i in range(4):
            nx, ny = x + dx[i], y + dy[i]
            if 0 <= ny < h+2 and 0 <= nx < w+2 and (ny, nx) not in inq and
matrix[ny][nx] == 0:
```

```

        if i != idx:
            q.append((ny, nx, segment + 1, i))
            inq.add((ny, nx))
        else:
            q.append((ny, nx, segment, idx))
            inq.add((ny, nx))

board = 0
while True:
    w, h = map(int, input().split())
    if w == 0 and h == 0:
        break
    matrix = [[0]*(w+2)]
    for _ in range(h):
        s = list(input())
        temp = [0]
        for si in s:
            if si == 'X':
                temp.append(1)
            else:
                temp.append(0)
        temp.append(0)
        matrix.append(temp)
    matrix.append([0]*(w+2))
    board += 1
    print('Board #{}:'.format(board))
    pair = 0
    while True:
        pair += 1
        ans = float('inf')
        x1, y1, x2, y2 = map(int, input().split())
        if x1 == x2 == y1 == y2 == 0:
            break
        matrix[y2][x2] = 0
        bfs(x1, y1, x2, y2)
        if ans != float('inf'):
            print('Pair {}: {} segments.'.format(pair, ans))
        else:
            print('Pair {}: impossible.'.format(pair))
        matrix[y2][x2] = 1
    print()

```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

源代码

```
#pylint:skip-file
from collections import deque
dx = [1, -1, 0, 0]
dy = [0, 0, 1, -1]
def bfs(x1, y1, x2, y2):
    global ans
    q = deque([(y1, x1, 0, -1)])
    inq = {(y1, x1)}
    while q:
        y, x, segment, idx = q.popleft()
        if y == y2 and x == x2:
            ans = min(ans, segment)
            continue
        for i in range(4):
            nx, ny = x + dx[i], y + dy[i]
            if 0 <= ny < h+2 and 0 <= nx < w+2 and (ny, nx) not in inq:
                if i != idx:
                    q.append((ny, nx, segment + 1, i))
                    inq.add((ny, nx))
                else:
                    q.append((ny, nx, segment, idx))
                    inq.add((ny, nx))

board = 0
while True:
    w, h = map(int, input().split())
    if w == 0 and h == 0:
        break
    matrix = [[0]*(w+2)]
    for _ in range(h):
        s = list(input())
        temp = [0]
        for si in s:
            if si == 'X':
                temp.append(1)
            else:
                temp.append(0)
```

基本信息

#: 47458436
题目: 02802
提交人: 24n2400011033
内存: 4024kB
时间: 56ms
语言: Python3
提交时间: 2024-11-29 11:33:52

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。

前两题是后面的铺垫，尤其看到flowers的状态转移方程，意识到这几题都很像。需要注意如何提高程序运行效率。

水淹七军那一题着实一直RE，后来发现water_height的m和n写反了，改了一下就AC，不知道为什么？

小游戏刚开始没看到提示，用dfs做了一遍，发现一直超时。改成bfs效率就高了很多。在一开始就应该选择正确的算法。