

1. (6min)

LuoguP1255 数楼梯

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

思路：显然为斐波那契数列

代码：

```
n=int(input())
N=max(n,2)
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])
```

运行：

**R192442670 记录详情**

编程语言	代码长度	用时	内存
Python 3	120B	154ms	4.99MB

测试点信息    源代码

测试点信息

#1 AC 15ms/3.60MB	#2 AC 15ms/3.64MB	#3 AC 15ms/3.64MB	#4 AC 15ms/3.69MB	#5 AC 15ms/3.72MB	#6 AC 15ms/3.68MB	#7 AC 15ms/3.63MB
#8 AC 16ms/3.77MB	#9 AC 16ms/4.07MB	#10 AC 17ms/4.99MB				

所属题目    P1255 数楼梯

评测状态    Accepted

评测分数    100

提交时间    2024-12-02 16:09:45

2. (5min)

27528: 跳台阶

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

思路：第 1 到  $n-1$  级台阶，每个台阶都可以选择经过或者不经过，共  $2^{(n-1)}$  种

代码：

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

运行:

#47516112提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: **Accepted**

源代码

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

基本信息

#: 47516112  
题目: 27528  
提交人: 24n2400011028  
内存: 4348kB  
时间: 40ms  
语言: Python3  
提交时间: 2024-12-02 16:15:22

©2002-2022 POJ 京ICP备20010980号-1

[English](#) [帮助](#) [关于](#)

3. (20min)

474D. Flowers

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

思路: dp, 递推即可

代码:

```
N=1e9+7
t,k=map(int,input().split())
da=[[0,0] for _ in range(t)]
ma=0
for i in range(t):
    da[i][0],da[i][1]=map(int,input().split())
    ma=max(ma,da[i][1])
dp=[1 for x in range(ma+1)]
num=[x for x in range(ma+1)]
for s in range(k,ma+1):
    dp[s]=(dp[s-1]+dp[s-k])%N
    num[s]=(num[s-1]+dp[s])%N
for p in range(t):
    print(int((num[da[p][1]]-num[da[p][0]-1])%N))
```

运行:

#	When	Who	Problem	Lang	Verdict	Time	Memory
294342801	Dec/02/2024 16:52 UTC+8	Torrential_WJP	474D - Flowers	Python 3	Accepted	562 ms	24100 KB

4. (25min)

LeetCode5.最长回文子串

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

思路：以字串的长度为指标从小到大 dp

代码：

```
class Solution:
    def longestPalindrome(self, s: str) -> str:
        n=len(s)
        start,maxlength=0,1
        dp=[[False]*n for _ in range(n)]
        for i in range(n):
            dp[i][i]=True
        for y in range(n-1):
            if s[y]==s[y+1]:
                dp[y][y+1]=True
                start,maxlength=y,2
        for x in range(2,n):
            for z in range(n-x):
                if s[z]==s[z+x] and dp[z+1][z+x-1]:
                    dp[z][z+x]=True
                    start,maxlength=z,x+1
        return(s[start:start+maxlength])
```

运行：

通过

Torrential 提交于 2024.12.03 16:17

官方题解 写题解

面向在校学生的专享优惠  
完成认证享 1 元/天升级 Plus 会员, 开启高效学习

执行用时分布  
1979 ms | 击败 54.01%

复杂度分析

消耗内存分布  
24.61 MB | 击败 11.00%

代码

```
Python3 智能模式
```

```
1 class Solution:
2     def longestPalindrome(self, s: str) -> str:
3         n=len(s)
4         start,maxlength=0,1
5         dp=[[False]*n for _ in range(n)]
6         for i in range(n):
7             dp[i][i]=True
8         for y in range(n-1):
9             if s[y]==s[y+1]:
10                 dp[y][y+1]=True
```

行 11, 列 17 | 已存储

测试用例 测试结果

通过 执行用时: 0 ms

Case 1 Case 2

输入

s =  
"cbdd"

5. (2h)

12029: 水淹七军

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

思路: 常规的 dfs, 但是 debug 搞了半天……最后发现是 break 的问题

代码:

```
from collections import deque
import sys
def bfs(x,y):
    global s11,s12,mat
    dir = [[0, -1], [0, 1], [1, 0], [-1, 0]]
    q=deque([(x,y)])
    while q:
        e,f=q.popleft()
        for dx,dy in dir:
            nx,ny=e+dx,f+dy
            if mat[e][f]>mat[nx][ny]:
                q.append((nx,ny))
                if nx==s11 and ny==s12:
                    return True
            mat[nx][ny]=mat[e][f]
    return False

lines=list(sys.stdin.read().split())
K=int(lines[0])
i=1
```


```

for _ in range(K):
    m,n=int(lines[i]),int(lines[i+1])
    i+=2
    mat=[[1025]*(n+2)]
    for a in range(m):

mat.append([1025]+list(map(int,lines[i:i+n]))+[1025
])
        i+=n
    mat.append([1025]*(n+2))
    s11,s12=int(lines[i]),int(lines[i+1])
    i+=2
    waternum=int(lines[i])
    i+=1
    answer=False
    for b in range(waternum):
        if bfs(int(lines[i]),int(lines[i+1])):
            answer=True
            i=i+2
    if answer:
        print('Yes')
    else:
        print('No')

```

运行：


**CS101 / 题库（包括计概、数算题目）**

[题目](#)
[排名](#)
[状态](#)
[提问](#)

#47538729提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: **Accepted**

源代码

```

from collections import deque
import sys
def bfs(x,y):
    global s11,s12,mat
    dir = [[0, -1], [0, 1], [1, 0], [-1, 0]]
    q=deque([(x,y)])
    while q:
        e,f=q.popleft()
        for dx,dy in dir:
            nx,ny=e+dx,f+dy
            if mat[e][f]>mat[nx][ny]:
                q.append((nx,ny))
                if nx==s11 and ny==s12:
                    return True
            mat[nx][ny]=mat[e][f]
    return False

lines=list(sys.stdin.read().split())
K=int(lines[0])
i=1
for _ in range(K):
    m,n=int(lines[i]),int(lines[i+1])

```

基本信息

#: 47538729  
 题目: 12029  
 提交人: 24n2400011028  
 内存: 5908kB  
 时间: 127ms  
 语言: Python3  
 提交时间: 2024-12-03 19:21:58

6. (1.5h)

02802: 小游戏

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

思路：dfs 时用指标标识上一次的方向，并且用 history 记录以某个特定方向到达某个点的最短线段数

代码：

```
from collections import deque
def bfs(x1,y1,x2,y2):
    global step
    minstep=6400
    dir=[[0,-1],[0,1],[1,0],[-1,0]]
    history={}
    q=deque([(x1,y1,-1,0)])
    while q:
        x,y,di,num= q.popleft()
        for i in range(4):
            dx,dy=dir[i][0],dir[i][1]
            nx,ny=x+dx,y+dy
            if i==di:
                newnum=num
            else:
                newnum=num+1
            if nx==x2 and ny==y2:
                minstep=min(minstep,newnum)
            if 0<=nx<=h+1 and 0<=ny<=w+1 and
mat[nx][ny]==0 and ((nx,ny,i) not in history or
history[(nx,ny,i)]>newnum):
                q.append((nx,ny,i,newnum))
                history[(nx, ny, i)] = newnum
        if minstep<6400:
            return(str(minstep)+' segments.')
        else:
            return('impossible.')

n=1
while True:
    w,h=map(int,input().split())
```

```

matstring=[]
if w==h==0:
    break
for _ in range(h):
    matstring.append(input())
mat=[[0]*(w+2) for _ in range(h+2)]
for i in range(h):
    for j in range(w):
        if matstring[i][j]=='X':
            mat[i+1][j+1]=1
ans='Board #'+str(n)+':\n'
n+=1
m=1
while True:
    x11,y11,x22,y22=map(int,input().split())
    if x11==y11==x22==y22==0:
        break
    ans+='Pair '+str(m)+':\n'+bfs(y11,x11,y22,x22)+'\n'
    m+=1
print(ans)

```

运行:


**CS101 / 题库 (包括计概、数算题目)**

[题目](#)
[排名](#)
[状态](#)
[提问](#)

---

**#47542709提交状态**

[查看](#)
[提交](#)
[统计](#)
[提问](#)

状态: **Accepted**

源代码
 

```

from collections import deque
def bfs(x1,y1,x2,y2):
    global step
    minstep=6400
    dir=[[0,-1],[0,1],[1,0],[-1,0]]
    history={}
    q=deque([(x1,y1,-1,0)])
    while q:
        x,y,di,num= q.popleft()
        for i in range(4):
            dx,dy=dir[i][0],dir[i][1]
            nx,ny=x+dx,y+dy
            if i==di:
                newnum=num
            else:
                newnum=num+1
            if nx==x2 and ny==y2:
                minstep=min(minstep,newnum)
            if 0<=nx<=h+1 and 0<=ny<=w+1 and mat[nx][ny]==0 and ((nx,ny) not in history):
                q.append((nx,ny,i,newnum))
                history[(nx,ny,i)] = newnum
    if minstep<6400:

```

基本信息
 

#: 47542709  
 题目: 02802  
 提交人: 24n2400011028  
 内存: 4740kB  
 时间: 85ms  
 语言: Python3  
 提交时间: 2024-12-03 22:44:21

总结与收获：

1. 巩固了 dfs 的栈写法，感觉这个方法似乎在某些方面更加普适一些，而且比较方便 return 回想要的值？
2. 学到了一些特殊的输入格式写法