

Assignment #6: Recursion and DP

2024 fall, Compiled by 吕金浩, 物理学院

1. 题目

sy119: 汉诺塔

recursion, <https://sunnywhy.com/sfbj/4/3/119>

思路：主打一个递推

代码：

```
def move(rst,start,target,middle):
    if rst==1:
        print(start+'->'+target)
        return
    move(rst-1,start,middle,target)
    print(start+'->'+target)
    move(rst-1,middle,target,start)
n=int(input())
print(2**n-1)
move(n,'A','C','B')
```

```
1  ✓ def move(rst, start, target, middle):
2  ✓     if rst==1:
3      |         print(start+'->'+target)
4      |         return
5      |         move(rst-1, start, middle, target)
6      |         print(start+'->'+target)
7      |         move(rst-1, middle, target, start)
8  n=int(input())
9  print(2**n-1)
10 move(n, 'A', 'C', 'B')
```

测试输入

提交结果

历史提交

完美通过

[查看题解](#)

100% 数据通过测试

运行时长: 0 ms

sy132: 全排列 I

recursion, <https://sunnywhy.com/sfbj/4/3/132>

思路：主打一个偷懒

代码：

```
from itertools import *
```

```
n=int(input())
```

```
lst=list(range(1,n+1))
```

```
for x in permutations(lst,n):
```

```
    print(' '.join([str(y) for y in x]))
```

```
1  from itertools import *
2  n=int(input())
3  lst=list(range(1,n+1))
4  for x in permutations(lst,n):
5      print(' '.join([str(y) for y in x]))
```

测试输入

提交结果

历史提交

完美通过

[查看题解](#)

100% 数据通过测试

运行时长: 0 ms

02945: 拦截导弹

dp, <http://cs101.openjudge.cn/2024fallroutine/02945>

思路：前段时间问了 AI 才会做的（哭

代码：

```
n=int(input())
```

```
if not n:
```

```
    print(0)
```

```
else:
```

```
    bombs = [int(x) for x in input().split()]
```

```
    longest = [1] * n
```

```

for i in range(1, n):
    for j in range(i):
        if bombs[i] <= bombs[j]:
            longest[i] = max(longest[i], longest[j] + 1)
        #longest[i] = max(longest[i], longest[i - 1])
print(max(longest))

```

#46835039提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: **Accepted**

源代码

```

n=int(input())
if not n:
    print(0)
else:
    bombs = [int(x) for x in input().split()]
    longest = [1] * n
    for i in range(1, n):
        for j in range(i):
            if bombs[i] <= bombs[j]:
                longest[i] = max(longest[i], longest[j] + 1)
        #longest[i] = max(longest[i], longest[i - 1])
    print(max(longest))

```

基本信息

#: 46835039
 题目: 02945
 提交人: lvjinhao
 内存: 3624kB
 时间: 34ms
 语言: Python3
 提交时间: 2024-10-30 16:00:31

23421: 小偷背包

dp, <http://cs101.openjudge.cn/practice/23421>

思路: 看了答案才会做的 (大哭)

代码: (对着答案写出来的)

```

n,b=map(int,input().split())
price=[0]+[int(x) for x in input().split()]
weight=[0]+[int(x) for x in input().split()]

```

```

bag_status=[]
for _ in range(n+1):
    bag_status.append([0]*(b+1))
#bag_status=[[0]*(b+1)]*(n+1)
for i in range(1,1+n):
    for j in range(1,b+1):

        if weight[i]>j:
            bag_status[i][j]=bag_status[i-1][j]
        else:
            bag_status[i][j]=max(bag_status[i-1][j], bag_status[i-1][j-weight[i]]+price[i])
print(bag_status[-1][-1])

```

状态: Accepted

源代码

```
n,b=map(int,input().split())
price=[0]+[int(x) for x in input().split()]
weight=[0]+[int(x) for x in input().split()]

bag_status=[]
for _ in range(n+1):
    bag_status.append([0]*(b+1))
#bag_status=[0]*(b+1)* (n+1)
for i in range(1,1+n):
    for j in range(1,b+1):

        if weight[i]>j:
            bag_status[i][j]=bag_status[i-1][j]
        else:
            bag_status[i][j]=max(bag_status[i-1][j], bag_status[i-1][j-weight[i]]+price[i])
print(bag_status[-1][-1])
```

基本信息

#: 46835274
题目: 23421
提交人: lvjinhao
内存: 3648kB
时间: 23ms
语言: Python3
提交时间: 2024-10-30 16:14:52

02754: 八皇后

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

思路: 暴力遍历, 然后检验, 中间有微小优化, 好像没有用到 dp 和递归?

代码: (删掉了注释掉的代码)

```
from itertools import *
perm=list(range(8))
oks=[]
for x in permutations(perm,8):
    if_ok=True
    for i in range(8):
        if not if_ok:
            break
        queen=x[i]
        for j in range(i+1,8):
            if abs(x[j]-queen)==j-i:
                if_ok=False
                break
    if if_ok:
        oks.append(''.join([str(j+1) for j in x]))
for _ in range(int(input())):
    print(oks[int(input())-1])
```

状态: Accepted

源代码

```
from itertools import *
perm=list(range(8))
oks=[]
for x in permutations(perm,8):
    if_ok=True
    for i in range(8):
        if not if_ok:
            break
        queen=x[i]
        for j in range(i+1,8):
            if abs(x[j]-queen)==j-i:
                if_ok=False
                break
    if if_ok:
        oks.append(''.join([str(j+1) for j in x]))
for _ in range(int(input())):
    print(oks[int(input())-1])
```

基本信息

#: 46835365

题目: 02754

提交人: lvjinhao

内存: 3540kB

时间: 94ms

语言: Python3

提交时间: 2024-10-30 16:18:13

189A. Cut Ribbon

brute force, dp 1300 <https://codeforces.com/problemset/problem/189/A>

代码:

```
t,a,b,c=map(int,input().split())
a,b,c=map(int,reversed(sorted([a,b,c])))
#print((a,b,c))
max_num={x:0 for x in range(1-a,t+1)}
for i in range(1,t+1):
    max_num[i]=max(max_num[i-a]+1,max_num[i-b]+1,max_num[i-c]+1)
    if max_num[i]==1:
        max_num[i]=0
        if i==a or i==b or i==c:
            max_num[i]=1

print(max_num[t])
```

General

#	Author	Problem	Lang	Verdict	Time	Memory	Sent	Judged		
288656334	Practice: goodgoodbrother	189A - 39	Python 3	Accepted	78 ms	680 KB	2024-10-29 05:05:27	2024-10-29 05:05:58	☆	Compare

→ Source

Copy

```
t,a,b,c=map(int,input().split())
a,b,c=map(int,reversed(sorted([a,b,c])))
#print((a,b,c))
max_num={x:0 for x in range(1-a,t+1)}
for i in range(1,t+1):
    max_num[i]=max(max_num[i-a]+1,max_num[i-b]+1,max_num[i-c]+1)
    if max_num[i]==1:
        max_num[i]=0
        if i==a or i==b or i==c:
            max_num[i]=1

print(max_num[t])
```

2. 学习总结和收获

零基础，近几天的每日选做，感觉不像一二十天前的那么好做了。dp 确实不那么容易自己想到，拦截导弹&小偷背包，我都是看了答案/问了 AI 后才会做的。不过 Cut Ribbon 是自己想出来的。递归感觉最开始没那么好理解，虽然汉诺塔疑似有点简单了。总之（如果）学会 dp 和递归，感觉思维会得到一个质的飞跃。希望这次月考至少做出 5 道题。