

# Assignment #A: 递归、田忌赛马

Updated 2355 GMT+8 Nov 4, 2025

2025 fall, Compiled by 贺桢羽、心理与认知科学学院

## 说明:

### 1. 解题与记录:

对于每一个题目, 请提供其解题思路(可选), 并附上使用Python或C++编写的源代码(确保已在OpenJudge, Codeforces, LeetCode等平台上获得Accepted)。请将这些信息连同显示“Accepted”的截图一起填写到下方的作业模板中。(推荐使用Typora <https://typoraio.cn> 进行编辑, 当然你也可以选择Word。)无论题目是否已通过, 请标明每个题目大致花费的时间。

2. 提交安排: \*\*提交时, 请首先上传PDF格式的文件, 并将.md或.doc格式的文件作为附件上传至右侧的“作业评论”区。确保你的Canvas账户有一个清晰可见的本人头像, 提交的文件为PDF格式, 并且“作业评论”区包含上传的.md或.doc附件。

3. **延迟提交:** 如果你预计无法在截止日期前提交作业, 请提前告知具体原因。这有助于我们了解情况并可能为你提供适当的延期或其他帮助。

请按照上述指导认真准备和提交作业, 以保证顺利完成课程要求。

## 1. 题目

### M018160: 最大连通域面积

dfs similar, <http://cs101.openjudge.cn/pctbook/M18160>

思路: 跟晶矿那道题做法类似, 用dfs。

代码

```
t=int(input())
for _ in range(t):
    n,m=map(int,input().split())
    ma=[["." for _ in range(m+2)]]
    for _ in range(n):
        a=["."]
        a.extend(list(input()))
        a.append(".")
        ma.append(a)
    ma.append(["." for _ in range(m+2)])
    di=[(0,1),(0,-1),(1,0),(-1,0),(1,-1),(1,1),(-1,-1),(-1,1)]
    def dfs(x,y):
        ma[x][y]="."
        for i in di:
            if ma[x+i[0]][y+i[1]]=="w":
```

```

        global count
        count+=1
        dfs(x+i[0],y+i[1])

res=[0]
for x in range(1,n+1):
    for y in range(1,m+1):
        if ma[x][y]=="W":
            count=1
            dfs(x,y)
            res.append(count)
print(max(res))

```

代码运行截图 (至少包含有"Accepted")

#50795258提交状态

View Submit Statistics Clarify

状态: Accepted

Source Code

```

t=int(input())
for _ in range(t):
    n,m=map(int,input().split())
    ma=[["." for _ in range(m+2)]]
    for _ in range(n):
        a=["."]
        a.extend(list(input()))
        a.append(".")
        ma.append(a)
    ma.append(["." for _ in range(m+2)])
    di=[(0,1),(0,-1),(1,0),(-1,0),(1,-1),(1,1),(-1,-1),(-1,1)]
    def dfs(x,y):
        ma[x][y]="."
        for i in di:
            if ma[x+i[0]][y+i[1]]=="W":
                global count
                count+=1
                dfs(x+i[0],y+i[1])

res=[0]
for x in range(1,n+1):
    for y in range(1,m+1):
        if ma[x][y]=="W":
            count=1
            dfs(x,y)
            res.append(count)
print(max(res))

```

基本信息

#: 50795258  
 题目: M18160  
 提交人: 25n2500013720  
 内存: 3708kB  
 时间: 72ms  
 语言: Python3  
 提交时间: 2025-11-11 17:08:44

©2002-2022 POJ 京ICP备20010980号-1

中文 Help About

## sy134: 全排列III 中等

<https://sunnywhy.com/sfbj/4/3/134>

思路：就是把全排列继续再筛选相同序列。

代码

```

n=int(input())
num=list(map(str,input().split()))
answer = []
cn = []
check=[]
def backtrack():

```

```

        if len(cn) == len(num):
            answer.append(cn[:])
            return
        for i in range(len(num)):
            if i not in check:
                cn.append(num[i])
                check.append(i)
                backtrack()
                cn.pop()
                check.pop()
    backtrack()
    res=[]
    for x in answer:
        if x not in res:
            res.append(x)
    for x in res:
        print(" ".join(x))

```

代码运行截图 (至少包含有"Accepted")

代码书写

Python

```

1  n=int(input())
2  num=list(map(str,input().split()))
3  answer = []
4  cn = []
5  check=[]
6  def backtrack():
7      if len(cn) == len(num):
8          answer.append(cn[:])
9          return
10     for i in range(len(num)):
11         if i not in check:
12             cn.append(num[i])
13             check.append(i)
14             backtrack()
15             cn.pop()
16             check.pop()
17 backtrack()
18 res=[]
19 for x in answer:
20     if x not in res:
21         res.append(x)
22 for x in res:
23     print(" ".join(x))
24
25

```

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 详情

运行时长: 754 ms

## sy136: 组合II 中等

<https://sunnywhy.com/sfbj/4/3/136>

给定一个长度为n的序列，其中有n个互不相同的正整数，再给定一个正整数k，求从序列中任选k个的所有可能结果。

思路：dfs

代码

```
n,k=map(int,input().split())
num=list(map(str,input().split()))
ans=[]
def dfs(idx,temp):
    if len(temp)==k:
        ans.append(temp[:])
        return
    if idx>n-1:
        return
    temp.append(num[idx])
    dfs(idx+1,temp)
    temp.pop()
    dfs(idx+1,temp)
dfs(0,[])
for i in ans:
    print(" ".join(list(map(str,i))))
```

代码运行截图 (至少包含有"Accepted")

完美通过

100% 数据通过测试

运行时长: 0 ms

语言: Python

```
1  n,k=map(int,input().split())
2  num=list(map(str,input().split()))
3  ans=[]
4  def dfs(idx,temp):
5      if len(temp)==k:
6          ans.append(temp[:])
7          return
8      if idx>n-1:
9          return
10     temp.append(num[idx])
11     dfs(idx+1,temp)
12     temp.pop()
13     dfs(idx+1,temp)
14 dfs(0,[])
15 for i in ans:
16     print(" ".join(list(map(str,i))))
17
```

## sy137: 组合III 中等

<https://sunnywhy.com/sfbj/4/3/137>

思路：跟组合二没有啥大差别吧，就是最后多了一个去重的步骤。

代码

```
n,k=map(int,input().split())
num=list(map(str,input().split()))
ans=[]
```

```
def dfs(idx,temp):
    if len(temp)==k:
        ans.append(temp[:])
        return
    if idx>n-1:
        return
    temp.append(num[idx])
    dfs(idx+1,temp)
    temp.pop()
    dfs(idx+1,temp)
dfs(0,[])
ans_=[list(item) for item in set(tuple(sub) for sub in ans)]
for i in ans_:
    print(" ".join(list(map(str,i))))
```

代码运行截图 (至少包含有"Accepted")

完美通过

100% 数据通过测试

运行时长: 0 ms

语言: Python

```
1  n,k=map(int,input().split())
2  num=list(map(str,input().split()))
3  ans=[]
4  def dfs(idx,temp):
5      if len(temp)==k:
6          ans.append(temp[:])
7          return
8      if idx>n-1:
9          return
10     temp.append(num[idx])
11     dfs(idx+1,temp)
12     temp.pop()
13     dfs(idx+1,temp)
14 dfs(0,[])
15 ans_=[list(item) for item in set(tuple(sub) for sub
16 for i in ans_:
17     print(" ".join(list(map(str,i))))
18
```

## M04123: 马走日

dfs, <http://cs101.openjudge.cn/pctbook/M04123>

思路：终于自己独立做出来一道递归加回溯的题目了，其间一直WA，TLE，最后也还是解决了。

代码

```
ans=0
def dfs(step,x,y):
    if step==m*n:
```

```

    global ans
    ans+=1
    return
for i in di:
    tx = x + i[0]
    ty = y + i[1]
    if 0<=tx<=n-1 and 0<=ty<=m-1:
        if ma[tx][ty]:
            ma[tx][ty]=0
            dfs(step+1,tx, ty)
            ma[tx][ty]=1
t=int(input())
for _ in range(t):
    ans=0
    n,m,x,y=map(int,input().split())
    ma=[[1 for _ in range(m)]for _ in range(n)]
    ma[x][y] = 0
    di=[(-1,2),(-2,1),(-2,-1),(-1,-2),(1,2),(2,1),(2,-1),(1,-2)]
    dfs(1,x,y)
    print(ans)

```

代码运行截图 (至少包含有"Accepted")

#50825097提交状态

查看 提交 统计 提问

状态: Accepted

源代码

```

ans=0
def dfs(step,x,y):
    if step==m*n:
        global ans
        ans+=1
        return
    for i in di:
        tx = x + i[0]
        ty = y + i[1]
        if 0<=tx<=n-1 and 0<=ty<=m-1:
            if ma[tx][ty]:
                ma[tx][ty]=0
                dfs(step+1,tx, ty)
                ma[tx][ty]=1
t=int(input())
for _ in range(t):
    ans=0
    n,m,x,y=map(int,input().split())
    ma=[[1 for _ in range(m)]for _ in range(n)]
    ma[x][y] = 0
    di=[(-1,2),(-2,1),(-2,-1),(-1,-2),(1,2),(2,1),(2,-1),(1,-2)]
    dfs(1,x,y)
    print(ans)

```

基本信息

#: 50825097  
 题目: M04123  
 提交人: 25n2500013720  
 内存: 3576kB  
 时间: 3331ms  
 语言: Python3  
 提交时间: 2025-11-13 17:04:01

©2002-2022 POJ 京ICP备20010980号-1

English 帮助 关于

## T02287: Tian Ji -- The Horse Racing

greedy, dfs <http://cs101.openjudge.cn/pctbook/T02287>

思路：双指针，仔细想想如果田忌最快的马不能比王最快的马快，或者田忌最慢的马不能比王最慢的马快，就是用田忌跑得最慢的马去抵消王跑得最快的马，另外只需要考虑马速度相同的情况就可以了。

代码

```
while 1:
    n=int(input())
    if n==0:
        break
    else:
        tian=list(map(int,input().split()))
        king=list(map(int,input().split()))
        tian.sort()
        king.sort()
        i=0
        j=n-1
        p=0
        q=n-1
        cnt=0
        while i<=j:
            if tian[j]>king[q]:
                cnt+=1
                j-=1
                q-=1
            elif tian[i]>king[p]:
                cnt+=1
                i+=1
                p+=1
            elif tian[i]==king[q]:
                q-=1
                i+=1
            else:
                cnt-=1
                q-=1
                i+=1
        print(cnt*200)
```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

源代码

```
from collections import deque
while 1:
    n=int(input())
    if n==0:
        break
    else:
        tian=list(map(int,input().split()))
        king=list(map(int,input().split()))
        tian.sort()
        king.sort()
        i=0
        j=n-1
        p=0
        q=n-1
        cnt=0
        while i<=j:
            if tian[j]>king[q]:
                cnt+=1
                j-=1
                q-=1
            elif tian[i]>king[p]:
                cnt+=1
                i+=1
                p+=1
            elif tian[i]==king[q]:
                q-=1
                i+=1
            else:
                cnt-=1
                q-=1
                i+=1
        print(cnt*200)
```

基本信息

#: 50892139  
题目: T02287  
提交人: 25n2500013720  
内存: 3868kB  
时间: 52ms  
语言: Python3  
提交时间: 2025-11-18 16:21:12

## 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2025fall每日选做”、CF、LeetCode、洛谷等网站题目。

上机课的模拟考试也太难了，根本下不了手。通过组合、全排列、最大连通域面积、马走日重新温习了dfs和回溯，本来以前一点也写不出来的算法，现在越来越熟练了。现在在补每日选做。