

# Assignment #4: Matrix, 链表 & Backtracking

Updated 2226 GMT+8 Sep 29, 2025

2025 fall, Compiled by 魏佳亮 物理学院

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

### E18161: 矩阵运算

matrices, <http://cs101.openjudge.cn/pctbook/E18161/>

请使用 @ 矩阵相乘运算符。

思路：implementation（好像openjudge用不了numpy库？）

代码：

```
d,e=map(int,input().split())
a=[]
for _ in range(d):
    a.append(list(map(int,input().split())))
f,g=map(int,input().split())
b=[]
for _ in range(f):
    b.append(list(map(int,input().split())))
h,i=map(int,input().split())
c=[]
```

```
for _ in range(h):
    c.append(list(map(int,input().split())))
if d==h and e==f and g==i:
    l=[[0 for _ in range(g)] for _ in range(d)]
    for i in range(d):
        for j in range(g):
            l[i][j]=c[i][j]+sum(a[i][k]*b[k][j] for k in range(e))
    for i in range(d):
        print(' '.join(str(x) for x in l[i]))
else:
    print('Error!')
```

代码运行截图 (至少包含有"Accepted")

#49944144提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: Accepted

源代码

```
d,e=map(int,input().split())
a=[]
for _ in range(d):
    a.append(list(map(int,input().split())))
f,g=map(int,input().split())
b=[]
for _ in range(f):
    b.append(list(map(int,input().split())))
h,i=map(int,input().split())
c=[]
for _ in range(h):
    c.append(list(map(int,input().split())))
if d==h and e==f and g==i:
    l=[[0 for _ in range(g)] for _ in range(d)]
    for i in range(d):
        for j in range(g):
            l[i][j]=c[i][j]+sum(a[i][k]*b[k][j] for k in range(e))
    for i in range(d):
        print(' '.join(str(x) for x in l[i]))
else:
    print('Error!')
```

基本信息

#: 49944144  
题目: E18161  
提交人: 2400011474  
内存: 5044kB  
时间: 73ms  
语言: Python3  
提交时间: 2025-09-07 16:57:39

## E19942: 二维矩阵上的卷积运算

matrices, <http://cs101.openjudge.cn/pctbook/E19942/>

思路: implementation

代码:

```

m,n,p,q=map(int,input().split())
t=[]
for _ in range(m):
    t.append(list(map(int,input().split())))
s=[]
for _ in range(p):
    s.append(list(map(int,input().split())))
l=[0]*(n+1-q) for _ in range(m+1-p)]
for i in range(m+1-p):
    for j in range(n+1-q):
        l[i][j]=sum(sum(s[k][h]*t[i+k][j+h] for h in range(q)) for k in range(p))
for i in range(m+1-p):
    print(' '.join(str(x) for x in l[i]))

```

代码运行截图 (至少包含有"Accepted")

#50193707提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: **Accepted**

源代码

```

m,n,p,q=map(int,input().split())
t=[]
for _ in range(m):
    t.append(list(map(int,input().split())))
s=[]
for _ in range(p):
    s.append(list(map(int,input().split())))
l=[0]*(n+1-q) for _ in range(m+1-p)]
for i in range(m+1-p):
    for j in range(n+1-q):
        l[i][j]=sum(sum(s[k][h]*t[i+k][j+h] for h in range(q)) for k in
for i in range(m+1-p):
    print(' '.join(str(x) for x in l[i]))

```

基本信息

#: 50193707  
 题目: E19942  
 提交人: 2400011474  
 内存: 3656kB  
 时间: 21ms  
 语言: Python3  
 提交时间: 2025-09-30 10:05:19

©2002-2022 POJ 京ICP备20010980号-1

[English](#) [帮助](#) [关于](#)

## M06640: 倒排索引

data structures, <http://cs101.openjudge.cn/pctbook/M06640/>

思路: dict

代码:

```

n=int(input())
l={}
for i in range(n):
    s=list(set(input().split()[1:]))
    for x in s:
        if x not in l:
            l[x]=[]

```

```

        l[x].append(str(i+1))
m=int(input())
for _ in range(m):
    x=input()
    if x in l:
        print(' '.join(l[x]))
    else:
        print('NOT FOUND')

```

代码运行截图 (至少包含有"Accepted")

#50193742提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: **Accepted**

源代码

```

n=int(input())
l={}
for i in range(n):
    s=list(set(input().split()[1:]))
    for x in s:
        if x not in l:
            l[x]=[]
            l[x].append(str(i+1))
m=int(input())
for _ in range(m):
    x=input()
    if x in l:
        print(' '.join(l[x]))
    else:
        print('NOT FOUND')

```

基本信息

#: 50193742  
 题目: M06640  
 提交人: 2400011474  
 内存: 9756kB  
 时间: 76ms  
 语言: Python3  
 提交时间: 2025-09-30 10:18:00

©2002–2022 POJ 京ICP备20010980号-1

[English](#) [帮助](#) [关于](#)

## E160.相交链表

two pointers, <https://leetcode.cn/problems/intersection-of-two-linked-lists/>

思路: yysy没想到 (包括下一道题), 感觉对链表的各种操作不大熟悉

代码:

```

class Solution:
    def getIntersectionNode(self, headA: ListNode, headB: ListNode) -> Optional[ListNode]:
        pA, pB = headA, headB
        while pA is not pB:
            pA = pA.next if pA else headB
            pB = pB.next if pB else headA
        return pA

```

代码运行截图 (至少包含有"Accepted")

← 全部提交记录

Python3 智能模式

通过 40 / 40 个通过的测试用例

Ama2ing Teresh... 提交于 2025.09.30 15:26

写题解

尊享面试 100 题

专为会员定制面试题单，涵盖完整知识架构与更多...

③ 执行用时分布

117 ms | 击败 10.89%

复杂度分析

③ 消耗内存分布

27.23 MB | 击败 62.01%

```
1 # Definition for singly-linked list.
2 # class ListNode:
3 #     def __init__(self, x):
4 #         self.val = x
5 #         self.next = None
6
7 class Solution:
8     def getIntersectionNode(self, headA: ListNode, headB: ListNode) -> Optional[ListNode]:
9         pA, pB = headA, headB
10        while pA is not pB:
11            pA = pA.next if pA else headB
12            pB = pB.next if pB else headA
13        return pA
14
```

E206.反转链表

three pointers, recursion, <https://leetcode.cn/problems/reverse-linked-list/>

思路：注意那三个的顺序（好难想到这种算法。。。可能还是不熟悉链表

代码

```
class Solution:
    def reverseList(self, head: Optional[ListNode]) -> Optional[ListNode]:
        pre,cur=None,head
        while cur:
            cur.next,pre,cur=pre,cur,cur.next
        return pre
```

(至少包含有"Accepted")

通过 28 / 28 个通过的测试用例

Ama2ing Teresh... 提交于 2025.09.30 15:30

写题解

尊享面试 100 题

专为会员定制面试题单，涵盖完整知识架构与更多...

③ 执行用时分布

0 ms | 击败 100.00%

复杂度分析

③ 消耗内存分布

18.50 MB | 击败 34.81%

```
1 # Definition for singly-linked list.
2 # class ListNode:
3 #     def __init__(self, val=0, next=None):
4 #         self.val = val
5 #         self.next = next
6 class Solution:
7     def reverseList(self, head: Optional[ListNode]) -> Optional[ListNode]:
8         pre,cur=None,head
9         while cur:
10             cur.next,pre,cur=pre,cur,cur.next
11        return pre
```

T02488: A Knight's Journey

backtracking, <http://cs101.openjudge.cn/practice/02488/>

思路：递归

代码

```
def found(s):
    if len(s)==p*q:
        return list(s)
    [x,y]=s[-1]
    d=[[-1,-2],[1,-2],[-2,-1],[2,-1],[-2,1],[2,1],[-1,2],[1,2]]
    for [dx,dy] in d:
        if 0<=x+dx<p and 0<=y+dy<q and l[x+dx][y+dy]==0:
            l[x+dx][y+dy]=1
            a=found(s+[[x+dx,y+dy]])
            if a:
                return a
            l[x+dx][y+dy]=0
    return

def solve(p,q):
    for j in range(q):
        for i in range(p):
            l[i][j]=1
            a=found([[i,j]])
            if a:
                return a
    return

n=int(input())
for i in range(1,n+1):
    p,q=map(int,input().split())
    l=[[0]*q for _ in range(p)]
    a=solve(p,q)
    if a:
        s=''
        for [x,y] in a:
            s+=chr(y+65)+str(x+1)
    else:
        s='impossible'
    print(f'Scenario #{i}:')
    if i==n:
        print(s)
    else:
        print(s+'\n')
```

(至少包含有"Accepted")

状态: **Accepted**

源代码

```
def found(s):
    if len(s)==p*q:
        return list(s)
    [x,y]=s[-1]
    d=[[-1,-2],[1,-2],[-2,-1],[2,-1],[-2,1],[2,1],[-1,2],[1,2]]
    for [dx,dy] in d:
        if 0<=x+dx<p and 0<=y+dy<q and l[x+dx][y+dy]==0:
            l[x+dx][y+dy]=1
            a=found(s+[[x+dx,y+dy]])
            if a:
                return a
            l[x+dx][y+dy]=0
    return
def solve(p,q):
    for j in range(q):
        for i in range(p):
            l[i][j]=1
            a=found([[i,j]])
            if a:
                return a
    return
n=int(input())
for i in range(1,n+1):
    p,q=map(int,input().split())
    l=[[0]*q for _ in range(p)]
    a=solve(p,q)
    if a:
        s=''
        for [x,y] in a:
            s+=chr(y+65)+str(x+1)
    else:
        s='impossible'
    print(f'Scenario #{i}:')
    if i==n:
        print(s)
    else:
        print(s+'\n')
```

基本信息

#: 49988238  
题目: 02488  
提交人: 2400011474  
内存: 3688kB  
时间: 233ms  
语言: Python3  
提交时间: 2025-09-14 12:14:24

## 2. 学习总结和个人收获

如果发现作业题目相对简单，有否寻找额外的练习题目，如“数算2025fall每日选做”、LeetCode、Codeforces、洛谷等网站上的题目。

依然跟着每日选做。。。由于在数算上花的时间不多，有点想看看月考会被干爆成什么样（