

# Assignment #B: 图 (1/4)

Updated 2031 GMT+8 Nov 17, 2025

2025 fall, Compiled by 同学的姓名、院系

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

### E07218: 献给阿尔吉侬的花束

bfs, <http://cs101.openjudge.cn/practice/07218/>

思路: 正常的bfs

代码:

```
from collections import deque
q=int(input())
for _ in range(q):
    r,c=map(int,input().split())
    l=[]
    for i in range(r):
        s=list(input())
        l.append(s)
        if 'S' in s:
            a=(i,s.index('S'),0)
            l[i][a[1]]='#'
        if 'E' in s:
```

```

        b=(i,s.index('E'))
t=deque([a])
p=((-1,0),(1,0),(0,-1),(0,1))
k=False
while t:
    x,y,z=t.popleft()
    for dx,dy in p:
        if 0<=x+dx<r and 0<=y+dy<c:
            if l[x+dx][y+dy]=='.':
                l[x+dx][y+dy]='#'
                t.append((x+dx,y+dy,z+1))
            elif l[x+dx][y+dy]=='E':
                print(z+1)
                k=True
                break
    if k:
        break
if not k:
    print('oop!')

```

代码运行截图 (至少包含有"Accepted")

#50894971提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: **Accepted**

源代码

```

from collections import deque
q=int(input())
for _ in range(q):
    r,c=map(int,input().split())
    l=[]
    for i in range(r):
        s=list(input())
        l.append(s)
        if 'S' in s:
            a=(i,s.index('S'),0)
            l[i][a[1]]='#'
        if 'E' in s:
            b=(i,s.index('E'))
t=deque([a])
p=((-1,0),(1,0),(0,-1),(0,1))
k=False
while t:
    x,y,z=t.popleft()
    for dx,dy in p:
        if 0<=x+dx<r and 0<=y+dy<c:
            if l[x+dx][y+dy]=='.':
                l[x+dx][y+dy]='#'
                t.append((x+dx,y+dy,z+1))
            elif l[x+dx][y+dy]=='E':
                print(z+1)
                k=True
                break
    if k:
        break
if not k:
    print('oop!')

```

基本信息

#: 50894971  
 题目: 07218  
 提交人: 24n2400011474  
 内存: 3824kB  
 时间: 97ms  
 语言: Python3  
 提交时间: 2025-11-18 19:21:56

## M27925: 小组队列

dict, queue, <http://cs101.openjudge.cn/practice/27925/>

思路：implementation，感觉中间插队不能用deque？

代码：

```
from collections import deque
t=int(input())
d={}
for i in range(t):
    l=input().split()
    for x in l:
        d[x]=i
q=[]
p=[]
while True:
    s=input().split()
    if s[0]=='STOP':
        break
    elif s[0]=='ENQUEUE':
        if s[1] not in d:
            q.append(s[1])
            p.append(-1)
        elif d[s[1]] not in p:
            q.append(s[1])
            p.append(d[s[1]])
        else:
            for i in range(len(q))[::-1]:
                if p[i]==d[s[1]]:
                    p=p[:i+1]+[d[s[1]]]+p[i+1:]
                    q=q[:i+1]+[s[1]]+q[i+1:]
                    break
    else:
        print(q[0])
        p=p[1:]
        q=q[1:]
```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

## 基本信息

#: 50895151

题目: 27925

提交人: 24n2400011474

内存: 5260kB

时间: 165ms

语言: Python3

提交时间: 2025-11-18 19:36:13

## 源代码

```
from collections import deque
t=int(input())
d={}
for i in range(t):
    l=input().split()
    for x in l:
        d[x]=i
q=[]
p=[]
while True:
    s=input().split()
    if s[0]=='STOP':
        break
    elif s[0]=='ENQUEUE':
        if s[1] not in d:
            q.append(s[1])
            p.append(-1)
        elif d[s[1]] not in p:
            q.append(s[1])
            p.append(d[s[1]])
        else:
            for i in range(len(q))[:-1]:
                if p[i]==d[s[1]]:
                    p=p[:i+1]+[d[s[1]]]+p[i+1:]
                    q=q[:i+1]+[s[1]]+q[i+1:]
                    break
            else:
                print(q[0])
                p=p[1:]
                q=q[1:]
```

## M04089: 电话号码

trie, <http://cs101.openjudge.cn/practice/04089/>

思路: sort

代码:

```
t=int(input())
for _ in range(t):
    n=int(input())
    l=[]
    for _ in range(n):
        l.append(input())
    l.sort()
    k=True
    for i in range(n-1):
        if len(l[i])<=len(l[i+1]) and l[i]==l[i+1][:len(l[i])]:
            k=False
            break
    if k:
```

```
print('YES')
else:
    print('NO')
```

代码运行截图 (至少包含有"Accepted")

#50747513提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: Accepted

源代码

```
t=int(input())
for _ in range(t):
    n=int(input())
    l=[]
    for _ in range(n):
        l.append(input())
    l.sort()
    k=True
    for i in range(n-1):
        if len(l[i])<=len(l[i+1]) and l[i]==l[i+1][:len(l[i])]:
            k=False
            break
    if k:
        print('YES')
    else:
        print('NO')
```

基本信息

#: 50747513  
题目: 04089  
提交人: 24n2400011474  
内存: 4296kB  
时间: 87ms  
语言: Python3  
提交时间: 2025-11-08 09:29:13

©2002-2022 POJ 京ICP备20010980号-1

[English](#) [帮助](#) [关于](#)

## M3532.针对图的路径存在性查询I

disjoint set, <https://leetcode.cn/problems/path-existence-queries-in-a-graph-i/>

思路：并查集

代码

```
class Solution:
    def pathExistenceQueries(self, n: int, nums: List[int], maxDiff: int, queries:
List[List[int]]) -> List[bool]:
        l=[i for i in range(n)]
        def find(x):
            if l[x]!=x:
                l[x]=find(l[x])
            return l[x]
        def union(x,y):
            a,b=find(x),find(y)
            l[b]=a
        for i in range(n-1):
            if nums[i+1]-nums[i]<=maxDiff:
                union(i,i+1)
```

```

s=[]
for x,y in queries:
    if find(x)==find(y):
        s.append(True)
    else:
        s.append(False)
return s

```

代码运行截图 (至少包含有"Accepted")

通过 550 / 550 个通过的测试用例

写题解

AmaZing TereshkovaXVM 提交于 2025.11.18 20:10

① 执行用时分布

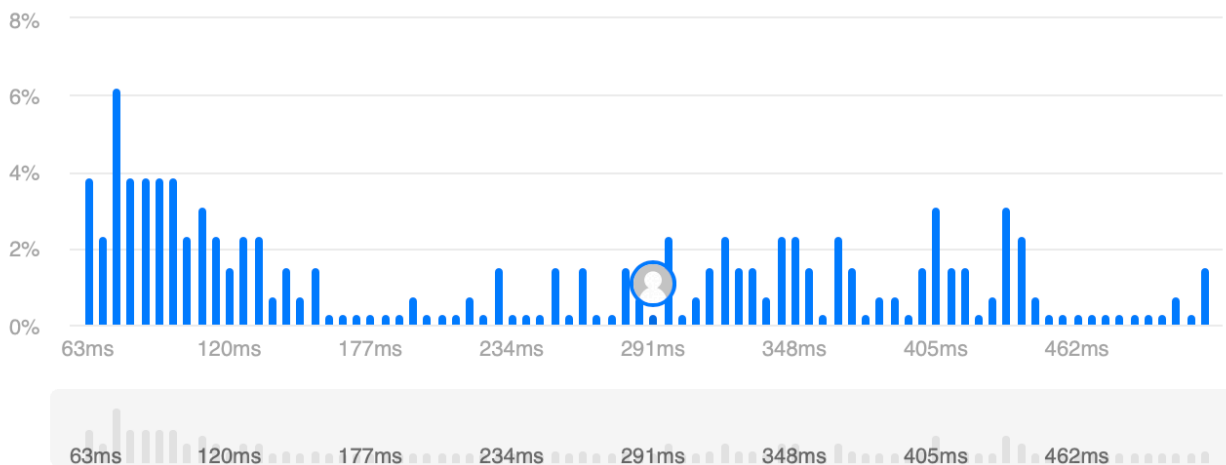


🗄 消耗内存分布

292 ms | 击败 44.96%

52.24 MB | 击败 6.98%

🌟 复杂度分析



代码 | Python3

```

class Solution:
    def pathExistenceQueries(self, n: int, nums: List[int], maxDiff: int, queries:
        l=[i for i in range(n)]
        def find(x):
            if l[x]!=x:
                l[x]=find(l[x])
            return l[x]
        def union(x,y):

```

## M19943: 图的拉普拉斯矩阵

OOP, graph, implementation, <http://cs101.openjudge.cn/pctbook/E19943/>

要求创建Graph, Vertex两个类, 建图实现。

思路：implement

代码

```
class Vertex:
    def __init__(self, i):
        self.id=i
        self.degree=0
        self.neighbors=set()
    def add(self,i):
        self.neighbors.add(i)
        self.degree+=1
class Graph:
    def __init__(self,n):
        self.list=[Vertex(i) for i in range(n)]
    def union(self,x,y):
        self.list[y].add(x)
        self.list[x].add(y)
n,m=map(int,input().split())
l=Graph(n)
for _ in range(m):
    x,y=map(int,input().split())
    l.union(x,y)
for i in range(n):
    p=[0]*n
    p[i]=l.list[i].degree
    for j in l.list[i].neighbors:
        p[j]=-1
    print(' '.join([str(x) for x in p]))
```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

源代码

```
class Vertex:
    def __init__(self, i):
        self.id=i
        self.degree=0
        self.neighbors=set()
    def add(self,i):
        self.neighbors.add(i)
        self.degree+=1
class Graph:
    def __init__(self,n):
        self.list=[Vertex(i) for i in range(n)]
    def union(self,x,y):
        self.list[y].add(x)
        self.list[x].add(y)
n,m=map(int,input().split())
l=Graph(n)
for _ in range(m):
    x,y=map(int,input().split())
    l.union(x,y)
for i in range(n):
    p=[0]*n
    p[i]=l.list[i].degree
    for j in l.list[i].neighbors:
        p[j]=-1
    print(' '.join([str(x) for x in p]))
```

基本信息

#: 50896017  
题目: E19943  
提交人: 24n2400011474  
内存: 3644kB  
时间: 22ms  
语言: Python3  
提交时间: 2025-11-18 20:28:01

## T25353: 排队

<http://cs101.openjudge.cn/pctbook/T25353/>

思路：一遍一遍筛，把能到最前面的排序

代码：

```
n,d=map(int,input().split())
l=[]
for _ in range(n):
    l.append(int(input()))
r=[]
t=[1 for _ in range(n)]
def solve(l,t):
    global r
    if t==[0]*len(l):
        return
    s=[]
    a=float('-inf')
    b=float('inf')
    for i in range(n):
        if t[i]:
            a=max(l[i],a)
```



```
        b=min(l[i],b)
        if l[i]>=a-d and l[i]<=b+d:
            s.append(l[i])
            t[i]=0
    r=r+sorted(s)
    solve(l,t)
solve(l,t)
for x in r:
    print(x)
```

代码运行截图 (至少包含有"Accepted")

#50842752提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: **Accepted**

源代码

```
n,d=map(int,input().split())
l=[]
for _ in range(n):
    l.append(int(input()))
r=[]
t=[1 for _ in range(n)]
def solve(l,t):
    global r
    if t==[0]*len(l):
        return
    s=[]
    a=float('-inf')
    b=float('inf')
    for i in range(n):
        if t[i]:
            a=max(l[i],a)
            b=min(l[i],b)
            if l[i]>=a-d and l[i]<=b+d:
                s.append(l[i])
                t[i]=0
    r=r+sorted(s)
    solve(l,t)
solve(l,t)
for x in r:
    print(x)
```

基本信息

#: 50842752  
题目: T25353  
提交人: 24n2400011474  
内存: 12212kB  
时间: 315ms  
语言: Python3  
提交时间: 2025-11-14 22:11:08

©2002-2022 POJ 京ICP备20010980号-1

[English](#) [帮助](#) [关于](#)

## 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。

之前的每日选做都补完了，希望考试的难度和知识范围能被每日选做完美地包含