

# Assignment #7: April 月考

2024 spring, Compiled by 胡豪俊 工学院

### 说明:

- 1) 请把每个题目解题思路 (可选), 源码Python, 或者C++ (已经在Codeforces/Openjudge上AC), 截图 (包含Accepted), 填写到下面作业模版中 (推荐使用 typora <https://typoraio.cn>, 或者用 word)。AC 或者没有AC, 都请标上每个题目大致花费时间。
- 2) 提交时候先提交pdf文件, 再把md或者doc文件上传到右侧“作业评论”。Canvas需要有同学清晰头像、提交文件有pdf、“作业评论”区有上传的md或者doc附件。
- 3) 如果不能在截止前提交作业, 请写明原因。

### 编程环境

操作系统: Windows11

Python编程环境: Visual Studio Code

## 1. 题目

### 27706: 逐词倒放

<http://cs101.openjudge.cn/practice/27706/>

代码

```
L=[str(i) for i in input().split()]
L.reverse()
print(" ".join(L))
##一脸懵逼地ac了
```

#### #44576005提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: Accepted

源代码

```
L=[str(i) for i in input().split()]
L.reverse()
print(" ".join(L))
```

基本信息

#: 44576005  
题目: 27706  
提交人: 22n2200011800  
内存: 3592kB  
时间: 22ms  
语言: Python3  
提交时间: 2024-04-08 19:15:39

## 27951: 机器翻译

<http://cs101.openjudge.cn/practice/27951/>

代码

```
q=[]
m,n=map(int,input().split())
L=[int(i) for i in input().split()]
ans=0
for i in L:
    if i not in q:
        if len(q)==m:
            q.pop()
            q.insert(0,i)
            ans+=1
        else:
            q.insert(0,i)
            ans+=1

print(ans)

##模拟题目中给出的操作即可，可以用队列实现，为了省时间直接用列表了
```

#44576262提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: **Accepted**

源代码

```
q=[]
m,n=map(int,input().split())
L=[int(i) for i in input().split()]
ans=0
for i in L:
    if i not in q:
        if len(q)==m:
            q.pop()
            q.insert(0,i)
            ans+=1
        else:
            q.insert(0,i)
            ans+=1

print(ans)
```

基本信息

#: 44576262  
题目: 27951  
提交人: 22n2200011800  
内存: 3644kB  
时间: 26ms  
语言: Python3  
提交时间: 2024-04-08 19:33:51

©2002-2022 POJ 京ICP备20010980号-1

[English](#) [帮助](#) [关于](#)

## 27932: Less or Equal

<http://cs101.openjudge.cn/practice/27932/>

代码

```
def merge_sort(array):
    if len(array)>1:
        middle=len(array)//2
        Left=array[:middle]
        Right=array[middle:]

        merge_sort(Left)
        merge_sort(Right)

    i=j=k=0
```

```

        while i<len(Left) and j<len(Right):
            if Left[i]<=Right[j]:
                array[k]=Left[i]
                i+=1
            else:
                array[k]=Right[j]
                j+=1
            k+=1

        while i<len(Left):
            array[k]=Left[i]
            i+=1
            k+=1
        while j<len(Right):
            array[k]=Right[j]
            j+=1
            k+=1
    return array

```

```

m,k=map(int,input().split())
L=[int(i) for i in input().split()]
merge_sort(L)
if k==m:
    print(L[-1])
elif k==0:
    if L[0]>1:
        print("1")
    else:
        print("-1")
else:
    if L[k-1]==L[k]:
        print("-1")
    else:
        print(L[k-1])

```

##先排序后找元素，这里为了省事直接用了merge\_sort，最后输出需要注意k=0和k=m的边界条件，自己写代码的时候总是只考虑一般情况而忽视极端情况，很不应该，在题目给出数据范围时就应该注意

状态: Accepted

源代码

```
def merge_sort(array):
    if len(array)>1:
        middle=len(array)//2
        Left=array[:middle]
        Right=array[middle:]

        merge_sort(Left)
        merge_sort(Right)

        i=j=k=0
        while i<len(Left) and j<len(Right):
            if Left[i]<=Right[j]:
                array[k]=Left[i]
                i+=1
            else:
                array[k]=Right[j]
                j+=1
            k+=1

        while i<len(Left):
            array[k]=Left[i]
            i+=1
            k+=1
        while j<len(Right):
            array[k]=Right[j]
            j+=1
            k+=1

        return array

m,k=map(int,input().split())
L=[int(i) for i in input().split()]
merge_sort(L)
if k==m:
    print(L[-1])
elif k==0:
    if L[0]>1:
        print("1")
    else:
        print("-1")
else:
    if L[k-1]==L[k]:
        print("-1")
    else:
        print(L[k-1])
```

基本信息

#: 44577208  
题目: 27932  
提交人: 22n2200011800  
内存: 9844kB  
时间: 267ms  
语言: Python3  
提交时间: 2024-04-08 20:33:24

## 27948: FBI树

<http://cs101.openjudge.cn/practice/27948/>

代码

```
class Node:
    def __init__(self):
        self.value=None
        self.left=None
        self.right=None

def build_fbi_tree(string):
    root=Node()
    if "0" not in string:
        root.value="I"
    elif "1" not in string:
        root.value="B"
    else:
        root.value="F"

    x=len(string)//2
    if x>0:
        root.left=build_fbi_tree(string[:x])
        root.right=build_fbi_tree(string[x:])
```

```

        return root

def post_traverse(node):
    ans=[]
    if node:
        ans.extend(post_traverse(node.left))
        ans.extend(post_traverse(node.right))
        ans.append(node.value)
    return "".join(ans)
n=input()
string=input()
root=build_fbi_tree(string)
print(post_traverse(root))
##套模板，熟悉树的写法之后还是能解决的。

```

## #44577715提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: **Accepted**

源代码

```

class Node:
    def __init__(self):
        self.value=None
        self.left=None
        self.right=None

def build_fbi_tree(string):
    root=Node()
    if "0" not in string:
        root.value="I"
    elif "1" not in string:
        root.value="B"
    else:
        root.value="F"

    x=len(string)//2
    if x>0:
        root.left=build_fbi_tree(string[:x])
        root.right=build_fbi_tree(string[x:])
    return root

def post_traverse(node):
    ans=[]
    if node:
        ans.extend(post_traverse(node.left))
        ans.extend(post_traverse(node.right))
        ans.append(node.value)
    return "".join(ans)
n=input()
string=input()
root=build_fbi_tree(string)
print(post_traverse(root))

```

基本信息

#: 44577715  
 题目: 27948  
 提交人: 22n2200011800  
 内存: 3832kB  
 时间: 28ms  
 语言: Python3  
 提交时间: 2024-04-08 21:05:53

©2002-2022 POJ 京ICP备20010980号-1

[English](#) [帮助](#) [关于](#)

## 27925: 小组队列

<http://cs101.openjudge.cn/practice/27925/>

代码

```

from collections import deque

t=int(input())
groups={}
member_to_group={}
for _ in range(t):
    members=list(map(int,input().split()))
    group_id=members[0]

```

```

groups[group_id]=deque()
for member in members:
    member_to_group[member]=group_id

queue=deque()

queue_set=set()

while True:
    command=input().split()
    if command[0]=='STOP':
        break
    elif command[0]=='ENQUEUE':
        x=int(command[1])
        group=member_to_group.get(x, None)
        if group is None:
            group=x
            groups[group]=deque([x])
            member_to_group[x]=group
        else:
            groups[group].append(x)
        if group not in queue_set:
            queue.append(group)
            queue_set.add(group)
    elif command[0]=='DEQUEUE':
        if queue:
            group=queue[0]
            x=groups[group].popleft()
            print(x)
            if not groups[group]:
                queue.popleft()
                queue_set.remove(group)

```

##参考了题解，这题在群里的讨论挺精彩的，对题目细节的把控需要十分注意，比如可能的散客，重复编号等等。

状态: Accepted

源代码

```
from collections import deque

t=int(input())
groups={}
member_to_group={}
for _ in range(t):
    members=list(map(int,input().split()))
    group_id=members[0]
    groups[group_id]=deque()
    for member in members:
        member_to_group[member]=group_id

queue=deque()
queue_set=set()

while True:
    command=input().split()
    if command[0]=='STOP':
        break
    elif command[0]=='ENQUEUE':
        x=int(command[1])
        group=member_to_group.get(x, None)
        if group is None:
            group=x
            groups[group]=deque([x])
            member_to_group[x]=group
        else:
            groups[group].append(x)
        if group not in queue_set:
            queue.append(group)
            queue_set.add(group)
    elif command[0]=='DEQUEUE':
        if queue:
            group=queue[0]
            x=groups[group].popleft()
            print(x)
            if not groups[group]:
                queue.popleft()
                queue_set.remove(group)
```

基本信息

#: 44579079  
题目: 27925  
提交人: 22n2200011800  
内存: 4964kB  
时间: 107ms  
语言: Python3  
提交时间: 2024-04-08 22:46:45

## 27928: 遍历树

<http://cs101.openjudge.cn/practice/27928/>

代码

```
def dfs(node,graph,result):
    if node not in graph:
        result.append(node)
        return
    children=graph[node]
    temp=[node]+children
    temp.sort()
    for child in temp:
        if child==node:
            result.append(node)
        elif child in graph:
            dfs(child,graph,result)

def main():
    n=int(input())
    graph={}
    all_nodes=set()
    child_nodes=set()
```

```

for _ in range(n):
    line=list(map(int, input().split()))
    node=line[0]
    all_nodes.add(node)
    if len(line)>1:
        children=line[1:]
        graph[node]=children
        child_nodes.update(children)
    else:
        graph[node]=[]

top_level_nodes=list(all_nodes-child_nodes)
top_level_nodes.sort()

result=[]
for node in top_level_nodes:
    dfs(node,graph,result)

for node in result:
    print(node)

if __name__ == "__main__":
    main()

```

##仍然参考了题解，还学习了其他同学的代码，遍历以前在计概都是硬写的很丑陋，现在应该能写得像模像样一点了

#### #44579148提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: **Accepted**

源代码

```

def dfs(node,graph,result):
    if node not in graph:
        result.append(node)
        return
    children=graph[node]
    temp=[node]+children
    temp.sort()
    for child in temp:
        if child==node:
            result.append(node)
        elif child in graph:
            dfs(child,graph,result)

def main():
    n=int(input())
    graph={}
    all_nodes=set()
    child_nodes=set()

    for _ in range(n):
        line=list(map(int, input().split()))
        node=line[0]
        all_nodes.add(node)
        if len(line)>1:
            children=line[1:]
            graph[node]=children
            child_nodes.update(children)
        else:
            graph[node]=[]

    top_level_nodes=list(all_nodes-child_nodes)
    top_level_nodes.sort()

    result=[]
    for node in top_level_nodes:
        dfs(node,graph,result)

    for node in result:
        print(node)

if __name__ == "__main__":
    main()

```

基本信息

#: 44579148  
 题目: 27928  
 提交人: 22n2200011800  
 内存: 3764kB  
 时间: 27ms  
 语言: Python3  
 提交时间: 2024-04-08 22:53:10



## 2. 学习总结和收获

---

期中真的很忙。笔试和月考没到场参加，自己后面找时间做了。笔试复习了一下感觉还行，月考这次在限定时间内大概能ac4到ac5的样子（小组队列看群里的讨论呆了很久才反应过来，做之前不该乱看的），总的来说感觉节奏把握得还行，树的部分还不是特别扎实，遇到一些需要手搓的可能在考场上会掉链子，还是要巩固好基础。