Assignment #D: May月考

Updated 1654 GMT+8 May 8, 2024

2024 spring, Complied by 王业成 生命科学学院

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

==（请改为同学的操作系统、编程环境等）==

操作系统：macOS Ventura 13.4.1 (c)

Python编程环境：Spyder IDE 5.2.2, PyCharm 2023.1.4 (Professional Edition)

C/C++编程环境：Mac terminal vi (version 9.0.1424), g++/gcc (Apple clang version 14.0.3, clang-1403.0.22.14.1)

1. 题目

02808: 校门外的树

http://cs101.openjudge.cn/practice/02808/

思路：列表模拟

代码

```
#
l,m=map(int,input().split())
lst=[0]*(l+1)
for i in range(m):
    a,b=map(int,input().split())
    lst[a:b+1]=[1]*(b-a+1)
num=0
for j in lst:
    if j==0:
        num+=1
print(num)
```

代码运行截图 ==（至少包含有"Accepted"）==

状态: Accepted

源代码

```
l,m=map(int,input().split())
lst=[0]*(l+1)
for i in range(m):
    a,b=map(int,input().split())
    lst[a:b+1]=[1]*(b-a+1)
num=0
for j in lst:
    if j==0:
        num+=1
print(num)
```

基本信息

#:	45024719
题目:	02808
提交人:	wangyecheng
内存:	3788kB
时间:	24ms
语言:	Python3
提交时间:	2024-05-20 14:21:50

20449: 是否被5整除

http://cs101.openjudge.cn/practice/20449/

思路:

代码

```
#
s=input()
result=[]
for i in range(0,len(s)):
    a=s[0:i+1]
    num=0
    for j in range(0,len(a)):
        if a[len(a)-1-j]=="1":
            num+=2**j
    if num%5==0:
        result.append("1")
    else:
        result.append("0")
print("".join(result))
```

代码运行截图 ==（至少包含有"Accepted"）==

状态: Accepted

源代码

```
s=input()
result=[]
for i in range(0,len(s)):
    a=s[0:i+1]
    num=0
    for j in range(0,len(a)):
        if a[len(a)-1-j]=="1":
            num+=2**j
    if num%5==0:
        result.append("1")
    else:
        result.append("0")
print("".join(result))
```

基本信息

#: 45024772
题目: 20449
提交人: wangyecheng
内存: 3616kB
时间: 24ms
语言: Python3
提交时间: 2024-05-20 14:28:04

01258: Agri-Net

http://cs101.openjudge.cn/practice/01258/

思路:

很好的prim模板题

代码

```
#
while True:
    try:
        n=int(input())
        graph=[]
        for _ in range(n):
            lst=list(map(int,input().split()))
            graph.append(lst)
        def prim(graph,n):
            dis=[0]*n
            flag=[False]*n
            flag[0]=True
            k=0
            for i in range(n):
                dis[i]=graph[k][i]
            for j in range(n-1):
                min=100000
                for i in range(n):
                    if min>dis[i] and not flag[i]:
                        min=dis[i]
                        k=i
                if k==0:
                    return
                flag[k]=True
                for i in range(n):
                    if dis[i]>graph[k][i] and not flag[i]:
                        dis[i]=graph[k][i]
            return dis
        a=prim(graph,n)
```

```
        num=0
        for i in a:
            num+=i
        print(num)
    except EOFError:
        break
```

代码运行截图 ==（AC代码截图，至少包含有"Accepted"）==

状态: Accepted

源代码

```
while True:
    try:
        n=int(input())
        graph=[]
        for _ in range(n):
            lst=list(map(int,input().split()))
            graph.append(lst)
        def prim(graph,n):
            dis=[0]*n
            flag=[False]*n
            flag[0]=True
            k=0
            for i in range(n):
                dis[i]=graph[k][i]
            for j in range(n-1):
                min=100000
                for i in range(n):
                    if min>dis[i] and not flag[i]:
                        min=dis[i]
                        k=i
                if k==0:
                    return
                flag[k]=True
                for i in range(n):
                    if dis[i]>graph[k][i] and not flag[i]:
                        dis[i]=graph[k][i]
            return dis
        a=prim(graph,n)
        num=0
        for i in a:
            num+=i
        print(num)
    except EOFError:
        break
```

基本信息

#: 45025371
题目: 01258
提交人: wangyecheng
内存: 4044kB
时间: 28ms
语言: Python3
提交时间: 2024-05-20 15:50:44

27635: 判断无向图是否连通有无回路(同23163)

http://cs101.openjudge.cn/practice/27635/

思路：起初不知道怎么同时判断回路和联通，学习了题解后豁然开朗

代码

```
#
n, m = list(map(int, input().split()))
edge = [[]for _ in range(n)]
for _ in range(m):
    a, b = list(map(int, input().split()))
    edge[a].append(b)
    edge[b].append(a)
```

```
cnt, flag = set(), False
def dfs(x, y):
    global cnt, flag
    cnt.add(x)
    for i in edge[x]:
        if i not in cnt:
            dfs(i, x)
        elif y != i:
            flag = True
for i in range(n):
    cnt.clear()
    dfs(i, -1)
    if len(cnt) == n:
        break
    if flag:
        break
print("connected:"+("yes" if len(cnt) == n else "no"))
print("loop:"+("yes" if flag else 'no'))
```

代码运行截图 ==（AC代码截图，至少包含有"Accepted"）==

状态: Accepted

源代码

```
n, m = list(map(int, input().split()))
edge = [[]for _ in range(n)]
for _ in range(m):
    a, b = list(map(int, input().split()))
    edge[a].append(b)
    edge[b].append(a)
cnt, flag = set(), False
def dfs(x, y):
    global cnt, flag
    cnt.add(x)
    for i in edge[x]:
        if i not in cnt:
            dfs(i, x)
        elif y != i:
            flag = True
for i in range(n):
    cnt.clear()
    dfs(i, -1)
    if len(cnt) == n:
        break
    if flag:
        break
print("connected:"+("yes" if len(cnt) == n else "no"))
print("loop:"+("yes" if flag else 'no'))
```

基本信息

#: 45025534
题目: 27635
提交人: wangyecheng
内存: 3844kB
时间: 29ms
语言: Python3
提交时间: 2024-05-20 16:09:58

27947: 动态中位数

http://cs101.openjudge.cn/practice/27947/

思路：用堆实现

代码

```
#
```

```
import heapq
def medium(nums):
    min_heap=[]
    max_heap=[]
    result=[]
    i=0
    for num in nums:
        i+=1
        if not max_heap or num<=-max_heap[0]:
            heapq.heappush(max_heap,-num)
        else:
            heapq.heappush(min_heap, num)
        if len(max_heap) - len(min_heap) > 1:
            heapq.heappush(min_heap, -heapq.heappop(max_heap))
        elif len(min_heap) > len(max_heap):
            heapq.heappush(max_heap, -heapq.heappop(min_heap))
        if i % 2 == 1:
            result.append(str(-max_heap[0]))
    return result
n=int(input())
for i in range(n):
    nums=list(map(int,input().split()))
    result=medium(nums)
    print(len(result))
    print(" ".join(result))
```

代码运行截图 ==（AC代码截图，至少包含有"Accepted"）==

状态: Accepted

源代码

```
import heapq
def medium(nums):
    min_heap=[]
    max_heap=[]
    result=[]
    i=0
    for num in nums:
        i+=1
        if not max_heap or num<=-max_heap[0]:
            heapq.heappush(max_heap,-num)
        else:
            heapq.heappush(min_heap, num)
        if len(max_heap) - len(min_heap) > 1:
            heapq.heappush(min_heap, -heapq.heappop(max_heap))
        elif len(min_heap) > len(max_heap):
            heapq.heappush(max_heap, -heapq.heappop(min_heap))
        if i % 2 == 1:
            result.append(str(-max_heap[0]))
    return result
n=int(input())
for i in range(n):
    nums=list(map(int,input().split()))
    result=medium(nums)
    print(len(result))
    print(" ".join(result))
```

基本信息

#:	45025786
题目:	27947
提交人:	wangyecheng
内存:	11796kB
时间:	288ms
语言:	Python3
提交时间:	2024-05-20 16:31:27

28190: 奶牛排队

http://cs101.openjudge.cn/practice/28190/

思路：

代码

```
#
N = int(input())
heights = [int(input()) for _ in range(N)]
left_bound = [-1] * N
right_bound = [N] * N
stack = []
for i in range(N):
    while stack and heights[stack[-1]] < heights[i]:
        stack.pop()
    if stack:
        left_bound[i] = stack[-1]
    stack.append(i)
stack = []
for i in range(N-1, -1, -1):
    while stack and heights[stack[-1]] > heights[i]:
        stack.pop()
    if stack:
        right_bound[i] = stack[-1]
    stack.append(i)
ans = 0
for i in range(N):
    for j in range(left_bound[i] + 1, i):
        if right_bound[j] > i:
            ans = max(ans, i - j + 1)
            break
print(ans)
```

代码运行截图 ==（AC代码截图，至少包含有"Accepted"）==

状态：Accepted

源代码

```python
N = int(input())
heights = [int(input()) for _ in range(N)]
left_bound = [-1] * N
right_bound = [N] * N
stack = []
for i in range(N):
    while stack and heights[stack[-1]] < heights[i]:
        stack.pop()
    if stack:
        left_bound[i] = stack[-1]
    stack.append(i)
stack = []
for i in range(N-1, -1, -1):
    while stack and heights[stack[-1]] > heights[i]:
        stack.pop()
    if stack:
        right_bound[i] = stack[-1]
    stack.append(i)
ans = 0
for i in range(N):
    for j in range(left_bound[i] + 1, i):
        if right_bound[j] > i:
            ans = max(ans, i - j + 1)
            break
print(ans)
```

基本信息

#：	45025841
题目：	28190
提交人：	wangyecheng
内存：	82456kB
时间：	2780ms
语言：	Python3
提交时间：	2024-05-20 16:36:52

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“2024spring每日选做”、CF、LeetCode、洛谷等网站题目。==

奶牛排队完全没思路，参考了题解学习了单调栈的写法，如果考试的话估计只能AC4，还要继续努力