

02808:校门外的树

```
def cs02808():  
    l, m = map(int, input().split())  
    lst = [True] * (l + 1)  
    for i in range(m):  
        x, y = map(int, input().split())  
        for j in range(x, y + 1):  
            lst[j] = False  
    print(lst.count(True))
```

#44901950提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: Accepted

源代码

```
def cs02808():  
    l, m = map(int, input().split())  
    lst = [True] * (l + 1)  
    for i in range(m):  
        x, y = map(int, input().split())  
        for j in range(x, y + 1):  
            lst[j] = False  
    print(lst.count(True))  
  
cs02808()
```

基本信息

#: 44901950
题目: 02808
提交人: 22n2200011358
内存: 3624kB
时间: 31ms
语言: Python3
提交时间: 2024-05-08 19:30:06

©2002-2022 POJ 京ICP备20010980号-1

[English](#) [帮助](#) [关于](#)

20449:是否被 5 整除

```
def cs20449():  
    ipt = input()  
    ans = ""  
    for i in range(len(ipt)):  
        if int(ipt[:i + 1], 2) % 5:  
            ans += "0"  
        else:
```

```
ans += "1"
```

```
print(ans)
```

#44903091提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: Accepted

源代码

```
def cs20449():
    ipt = input()
    ans = ""
    for i in range(len(ipt)):
        if int(ipt[:i + 1], 2) % 5:
            ans += "0"
        else:
            ans += "1"
    print(ans)

cs20449()
```

基本信息

#: 44903091
题目: 20449
提交人: 22n2200011358
内存: 3556kB
时间: 19ms
语言: Python3
提交时间: 2024-05-08 20:41:47

©2002-2022 POJ 京ICP备20010980号-1

[English](#) [帮助](#) [关于](#)

01258:Agri-Net

```
def cs01258():
```

```
    try:
```

```
        while True:
```

```
            n = int(input())
```

```
            a = []
```

```
            flag = [[True] * n] * n
```

```
            past = [0]
```

```
            ans = 0
```

```
            for i in range(n):
```

```
                a.append(list(map(int, input().split())))
```

```
            for i in range(n - 1):
```

```
                mn = 100001
```

```
                for j in past:
```

```

        for k in range(n):
            if flag[j][k] and j - k and
a[j][k] < mn:
                jj, kk, mn = j, k, a[j][k]
            flag[jj][kk], flag[kk][jj] = False, False
        ans += mn
    past.append(kk)
print(ans)
except:
    pass

```

#44910560提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: **Accepted**

源代码

```

def cs01258():
    try:
        while True:
            n = int(input())
            a = []
            flag = [[True] * n] * n
            past = [0]
            ans = 0
            for i in range(n):
                a.append(list(map(int, input().split())))
            for i in range(n - 1):
                mn = 100001
                for j in past:
                    for k in range(n):
                        if flag[j][k] and j - k and a[j][k] < mn:
                            jj, kk, mn = j, k, a[j][k]
                    flag[jj][kk], flag[kk][jj] = False, False
                ans += mn
            past.append(kk)
            print(ans)
    except:
        pass

cs01258()

```

基本信息

#: 44910560
 题目: 01258
 提交人: 22n2200011358
 内存: 4028kB
 时间: 99ms
 语言: Python3
 提交时间: 2024-05-09 17:46:42

©2002-2022 POJ 京ICP备20010980号-1

[English](#) [帮助](#) [关于](#)

27635:判断无向图是否连通有无回路(同 23163)

```
def cs27635():
```

```
    n, m = map(int, input().split())
```

```
dct1 = {i: [] for i in range(n)}
```

```
flag = [True for i in range(n)]
```

```
for i in range(m):
```

```
    u, v = map(int, input().split())
```

```
    dct1[u].append(v)
```

```
    dct1[v].append(u)
```

```
def go1(i):
```

```
    flag[i] = False
```

```
    for j in dct1[i]:
```

```
        if flag[j]:
```

```
            go1(j)
```

```
go1(0)
```

```
if True in flag:
```

```
    print("connected:no")
```

```
else:
```

```
    print("connected:yes")
```

```
flag = [True for i in range(n)]
```

```
def go2(ii, tmp):
```

```
    if flag[ii]:
```

```
flag[ii] = False
```

```
for j in dct1[ii]:
```

```
    if flag[j]:
```

```
        go2(j, tmp + [j])
```

```
    elif len(tmp) > 1 and j == tmp[-2]:
```

```
        pass
```

```
    else:
```

```
        print("loop:yes")
```

```
        exit()
```

```
for i in range(n):
```

```
    go2(i, [i])
```

```
print("loop:no")
```

状态: Accepted

源代码

```
def cs27635():
    n, m = map(int, input().split())
    dct1 = {i: [] for i in range(n)}
    flag = [True for i in range(n)]
    for i in range(m):
        u, v = map(int, input().split())
        dct1[u].append(v)
        dct1[v].append(u)

    def go1(i):
        flag[i] = False
        for j in dct1[i]:
            if flag[j]:
                go1(j)

    go1(0)
    if True in flag:
        print("connected:no")
    else:
        print("connected:yes")
    flag = [True for i in range(n)]

    def go2(ii, tmp):
        if flag[ii]:
            flag[ii] = False
            for j in dct1[ii]:
                if flag[j]:
                    go2(j, tmp + [j])
            elif len(tmp) > 1 and j == tmp[-2]:
                pass
            else:
                print("loop:yes")
                exit()

    for i in range(n):
        go2(i, [i])
    print("loop:no")
```

基本信息

#: 44904315
题目: 27635
提交人: 22n2200011358
内存: 3716kB
时间: 29ms
语言: Python3
提交时间: 2024-05-08 22:31:48

27947:动态中位数

```
def cs27947():
```

```
    import heapq
```

```
    for i in range(int(input())):
```

```
        ipt = list(map(int, input().split()))
```

```
        flag = 1
```

```
        lft, rt, ans = [], [], []
```

```
        md = ipt[0]
```

```
        f = 0
```

```
        for j in ipt:
```

```
        if j <= md:
            heapq.heappush(lft, -j)
            f += 1
        else:
            heapq.heappush(rt, j)
            f -= 1
        if f > 1:
            heapq.heappush(rt, -
            heapq.heappop(lft))
            f -= 2
        elif f < 0:
            heapq.heappush(lft, -
            heapq.heappop(rt))
            f += 2
        md = -lft[0]
        if f % 2:
            ans.append(md)
        flag += 1
    print(len(ans))
    for j in ans[::-1]:
        print(j, end=" ")
    print(ans[-1])
```

状态: Accepted

源代码

```
def cs27947():
    import heapq
    for i in range(int(input())):
        ipt = list(map(int, input().split()))
        flag = 1
        lft, rt, ans = [], [], []
        md = ipt[0]
        f = 0
        for j in ipt:
            if j <= md:
                heapq.heappush(lft, -j)
                f += 1
            else:
                heapq.heappush(rt, j)
                f -= 1
            if f > 1:
                heapq.heappush(rt, -heapq.heappop(lft))
                f -= 2
            elif f < 0:
                heapq.heappush(lft, -heapq.heappop(rt))
                f += 2
            md = -lft[0]
            if f % 2:
                ans.append(md)
            flag += 1
        print(len(ans))
        for j in ans[:-1]:
            print(j, end=" ")
        print(ans[-1])
```

cs27947()

基本信息

#: 44926536
题目: 27947
提交人: 22n2200011358
内存: 11572kB
时间: 259ms
语言: Python3
提交时间: 2024-05-11 00:52:27

28190:奶牛排队

```
def cs28190():
```

```
    n = int(input())
```

```
    a = [int(input()) for _ in range(n)]
```

```
    lft, rt = [n] * n, [-1] * n
```

```
    ans = 0
```

```
    lst1, lst2 = [], []
```

```
    for i in range(n):
```

```
        while lst1 and a[lst1[-1]] < a[i]:
```

```
            lst1.pop()
```

```
        if lst1:
```

```
            rt[i] = lst1[-1]
```



```
lst1.append(i)
```

```
while lst2 and a[lst2[-1]] > a[n - 1 - i]:
```

```
    lst2.pop()
```

```
if lst2:
```

```
    lft[n - 1 - i] = lst2[-1]
```

```
    lst2.append(n - 1 - i)
```

```
for i in range(n):
```

```
    for j in range(lft[i] - 1, i, -1):
```

```
        if rt[j] < i:
```

```
            ans = max(ans, j - i + 1)
```

```
            break
```

```
if ans > 1:
```

```
    print(ans)
```

```
else:
```

```
    print(0)
```

状态: **Accepted**

源代码

```
def cs28190():
    n = int(input())
    a = [int(input()) for _ in range(n)]
    lft, rt = [n] * n, [-1] * n
    ans = 0
    lst1, lst2 = [], []
    for i in range(n):
        while lst1 and a[lst1[-1]] < a[i]:
            lst1.pop()
        if lst1:
            rt[i] = lst1[-1]
        lst1.append(i)
        while lst2 and a[lst2[-1]] > a[n - 1 - i]:
            lst2.pop()
        if lst2:
            lft[n - 1 - i] = lst2[-1]
        lst2.append(n - 1 - i)
    for i in range(n):
        for j in range(lft[i] - 1, i, -1):
            if rt[j] < i:
                ans = max(ans, j - i + 1)
                break
    if ans > 1:
        print(ans)
    else:
        print(0)

cs28190()
```

基本信息

#: 44928229
题目: 28190
提交人: 22n2200011358
内存: 82492kB
时间: 2477ms
语言: Python3
提交时间: 2024-05-11 13:30:27

总结: 2 小时 AC4。T1T2 超出我的能力了。动态中位数改测试数据前, 我写的 $O(n^2)$ 还能过, 改数据之后我是看题解才明白这两题算法思路的。