

Assignment #5: cs201 Mock Exam

寒露第三天

Updated 1913 GMT+8 Oct 10, 2025

2025 fall, Complied by 杨浩、化院

1. 1. 题目

1.1 E29952: 咒语序列

Stack, <http://cs101.openjudge.cn/practice/29952/>

思路:

- 栈+动规

代码:

```

1  stack=[]
2  s=input()
3  dic={')':'(','}':'{',']':'['}
4  dp=[0]*len(s)
5  for i in range(len(s)):
6      if s[i]== '(':
7          stack.append('(')
8      elif s[i]== '[':
9          stack.append('[')
10     elif s[i]== '{':
11         stack.append('{')
12     else:
13         if not stack:
14             continue
15         if stack[-1]==dic[s[i]]:
16             stack.pop()
17             dp[i]=dp[i-1]+2
18             if i-dp[i]>=0:
19                 dp[i]+=dp[i-dp[i]]
20 print(max(dp))

```

Fence 1

代码运行截图 (至少包含有"Accepted")

#50297371提交状态

查看 提交 统计 提问

状态: Accepted

源代码

```
stack=[]
s=input()
dic={')': '(', '}': '{', ']': '['}
dp=[0]*len(s)
for i in range(len(s)):
    if s[i]==')':
        if dp[i]==0:
            stack.append(i)
        else:
            dp[i]=dp[stack.pop()]
    elif s[i]=='}':
        if dp[i]==0:
            stack.append(i)
        else:
            dp[i]=dp[stack.pop()]
    elif s[i]==']':
        if dp[i]==0:
            stack.append(i)
        else:
            dp[i]=dp[stack.pop()]
if len(stack)>0:
    print('No')
else:
    print('Yes')
```

基本信息

#: 50297371
题目: E29952
提交人: 25n2400011769
内存: 4432kB
时间: 32ms
语言: Python3
提交时间: 2025-10-10 18:42:49

Figure 1

1.2 M01328: Radar Installation

greedy, <http://cs101.openjudge.cn/practice/01328/>

思路:

- 贪心

代码:

```
1 num=0
2 while True:
3     n,d=map(int,input().split())
4     num +=1
5     if n==0 and d==0:
6         break
7     alist=[]
8     blist=[]
9     maxi=0
10    for i in range(n):
11        alist.append(tuple(map(int,input().split())))
12        maxi=max(maxi,alist[-1][1])
13    if maxi>d:
14        res=-1
15    else:
16        for x,y in alist:
17            r=(d*d-y*y)**0.5
18            blist.append((x-r,x+r))
19        blist.sort(key=lambda x:x[1])
20        res=1
21        begin=blist[0][1]
22        for i in blist:
23            if i[0]>begin:
24                res +=1
25                begin=i[1]
26
27    print(f'Case {num}: {res}')
28    input()
```

Fence 2

代码运行截图 (至少包含有"Accepted")

#50296658提交状态

查看 提交 统计 提问

状态: Accepted

基本信息

#:	50296658
题目:	M01328
提交人:	25n2400011769
内存:	3792kB
时间:	46ms
语言:	Python3
提交时间:	2025-10-10 18:18:25

```
源代码
num=0
while True:
    n,d=map(int,input().split())
    num +=1
    if n==0 and d==0:
        break
    alist=[]

```

Figure 2

1.3 M02754: 八皇后

dfs, <http://cs101.openjudge.cn/practice/02754/>

思路:

- 回溯

代码:

```
1 def dfs(deep):
2     if deep==8:
3         s=''.join(map(str,y))
4         s=int(s)
5         ans.append(s)
6         return
7     for k in range(1,9):
8         if k in y:
9             continue
10        else:
11            judge=True
12            dj=0
13            dk=1
14            for t in range(len(y)):
15                dj=t-(deep)
16                dk=y[t]-k
17                if dj==dk or dj+dk==0:
18                    judge=False
19            if judge:
20                y.append(k)
21                dfs(deep+1)
22                y.pop()
23
24 ans=[]
25 y=[]
26 dfs(0)
27 n=int(input())
28 for i in range(n):
29     a=int(input())
```

```
30 |     print(ans[a-1])
|_ Fence 3
```

代码运行截图 (至少包含有"Accepted")

#50296276提交状态

查看 提交 统计 提问

状态: Accepted

源代码

```
def dfs(deep):
    if deep==8:
        s=''.join(map(str,y))
        s=int(s)
        ans.append(s)
        return
    for k in range(1,9):
        ...
```

基本信息

#: 50296276
题目: M02754
提交人: 25n2400011769
内存: 3652kB
时间: 25ms
语言: Python3
提交时间: 2025-10-10 18:08:11

Figure 3

1.4 M25570: 洋葱

matrices, <http://cs101.openjudge.cn/practice/25570/>

思路:

代码:

```
1 n=int(input())
2 l=[]
3 for i in range(n):
4     l.append(list(map(int,input().split())))
5 s=0
6 if n!=1:
7     s +=sum(l[0])
8     s +=sum(l[-1])
9     for i in range(1,n-1):
10         s=s+l[i][0]+l[i][-1]
11 maxi=s
12 if n%2==0:
13     for i in range(1,n//2):
14         s=0
15         s +=sum(l[i][i:-i])
16         s +=sum(l[-1-i][i:-i])
17         for j in range(i+1,n-i-1):
18             s +=l[j][i]
19             s +=l[j][-i-1]
20         maxi=max(maxi,s)
21 else:
22     for i in range(1,n//2):
23         s=0
24         s +=sum(l[i][i:-i])
25         s +=sum(l[-1-i][i:-i])
```

```

26         for j in range(i+1,n-i-1):
27             s += l[j][i]
28             s += l[j][-i-1]
29             maxi=max(maxi,s)
30             maxi=max(maxi,l[(n)//2][(n)//2])
31
32     print(maxi)

```

Fence 4

代码运行截图 (至少包含有"Accepted")

#50297165提交状态

查看 提交 统计 提问

状态: Accepted

基本信息

#:	50297165
题目:	M25570
提交人:	25n2400011769
内存:	4000KB
时间:	21ms
语言:	Python3
提交时间:	2025-10-10 18:35:36

源代码

```

n=int(input())
l=[]
for i in range(n):
    l.append(list(map(int,input().split())))
s=0
if n!=1:
    s+=sum(l[0])
    s+=sum(l[-1])

```

Figure 4

1.5 M29954: 逃离紫罗兰监狱

bfs, <http://cs101.openjudge.cn/practice/29954/>

思路:

- bfs+集合优化队列时间和内存

代码

```

1  from collections import deque
2  def xieru(i,j,k,steps):
3      if l[i][j]=='.':
4          if (i,j,k) not in finishi_set:
5              duilie.append((i,j,k,steps+1))
6              finishi_set.add((i,j,k))
7      if l[i][j]=='#' and k>0:
8          if (i,j,k-1) not in finishi_set:
9              duilie.append((i,j,k-1,steps+1))
10             finishi_set.add((i,j,k-1))
11      if l[i][j]=='E':
12          return True
13      return False
14
15
16  r,c,k=map(int,input().split())
17  l=[]

```

```

18     for i in range(r):
19         l.append(input())
20         for j in range(c):
21             if l[-1][j]=='S':
22                 begin=(i,j)
23         duilie=deque([(begin[0],begin[1],k,0)])
24         finishi_set={(begin[0],begin[1],k)}
25         huitou_set=set()
26         delta=[(0,1),(0,-1),(1,0),(-1,0)]
27         while duilie:
28             t=duilie.popleft()
29             huitou_set.add((t[0],t[1]))
30             for di,dj in delta:
31                 if 0<=di+t[0]<r and 0<=dj+t[1]<c:
32                     if (di+t[0],dj+t[1]) not in huitou_set:
33                         if xieru(di+t[0],dj+t[1],t[2],t[3]):
34                             print(t[3]+1)
35                             exit()
36
37     print(-1)

```

Figure 5

(至少包含有"Accepted")

#50315475提交状态

查看 提交 统计 提问

状态: Accepted

源代码

```

from collections import deque
def xieru(i,j,k,steps):
    if l[i][j]=='.':
        if (i,j,k) not in finishi_set:
            duilie.append((i,j,k,steps+1))
            finishi_set.add((i,j,k))
    if l[i][j]=='#' and k>0:
        if (i,j,k-1) not in finishi_set:
            duilie.append((i,j,k-1,steps+1))

```

基本信息

#: 50315475
 题目: 29954
 提交人: 25n2400011769
 内存: 11428kB
 时间: 259ms
 语言: Python3
 提交时间: 2025-10-11 20:15:49

Figure 5

1.6 T27256: 当前队列中位数

backtracking, <http://cs101.openjudge.cn/practice/27256/>

思路:

- 大小堆+惰性删除

代码

```

1 import heapq
2 class zhongwushu():
3     def __init__(self):
4         self.small_size=0
5         self.big_size=0

```

```

6     def pingdui(self):
7         while self.small_size>self.big_size+1:
8             self.weihuduiding()
9             t=heapq.heappop(small_heap)
10            heapq.heappush(big_heap,-t)
11            self.small_size -=1
12            self.big_size +=1
13            self.weihuduiding()
14        while self.small_size<self.big_size:
15            self.weihuduiding()
16            t=heapq.heappop(big_heap)
17            heapq.heappush(small_heap,-t)
18            self.small_size +=1
19            self.big_size -=1
20            self.weihuduiding()
21    def weihuduiding(self):
22        while small_heap:
23            t=heapq.heappop(small_heap)
24            if -t in laze_dic:
25                laze_dic[-t] -=1
26                if laze_dic[-t]==0:
27                    laze_dic.pop(-t)
28            else:
29                heapq.heappush(small_heap,t)
30                break
31        while big_heap:
32            t=heapq.heappop(big_heap)
33            if t in laze_dic:
34                laze_dic[t] -=1
35                if laze_dic[t]==0:
36                    laze_dic.pop(t)
37            else:
38                heapq.heappush(big_heap,t)
39                break
40    n=int(input())
41    l=[]
42    big_heap=[]
43    small_heap=[]
44    laze_dic={}
45    deadline=0
46    m=zhongwushu()
47    for i in range(n):
48        ru=input()
49        if ru[0]=='a':
50            l.append(int(ru[4:]))
51            if m.small_size==0:
52                heapq.heappush(small_heap,-l[-1])
53                m.small_size +=1
54                m.pingdui()
55                continue
56            if l[-1]<=-small_heap[0]:

```

```

57         heapq.heappush(small_heaq,-l[-1])
58         m.small_size +=1
59     else:
60         heapq.heappush(big_heaq,l[-1])
61         m.big_size +=1
62         m.pingdui()
63     elif ru[0]=='d':
64         laze_dic.setdefault(l[deadline],0)
65         laze_dic[l[deadline]] +=1
66         if l[deadline]>-small_heaq[0]:
67             m.big_size -=1
68         else:
69             m.small_size -=1
70         m.pingdui()
71         m.weihuoding()
72         deadline +=1
73     else:
74         if m.small_size==m.big_size:
75             mid=(big_heaq[0]-small_heaq[0])/2
76             if mid==int(mid):
77                 mid=int(mid)
78             else:
79                 mid=-small_heaq[0]
80             print(mid)

```

Fence 6

(至少包含有"Accepted")

#50303306提交状态

查看 提交 统计 提问

状态: Accepted

基本信息

#:	50303306
题目:	T27256
提交人:	25n2400011769
内存:	8548kB
时间:	465ms
语言:	Python3
提交时间:	2025-10-10 22:00:10

源代码

```

import heapq
class zhongwushu():
    def __init__(self):
        self.small_size=0
        self.big_size=0
    def pingdui(self):
        while self.small_size>self.big_size+1:

```

Figure 6

2. 2. 学习总结和个人收获

月考的时候上来被第一题卡了一会，虽然是做过的题，但考试时动辄写错了，还好后来修正过来了。后续的雷达，八皇后以及中位数都是DSA中有过的题，前两个都可以较快的写出来，最后一个书写量太大了，考场上没写得下来。监狱和传统的bfs有点不同，直接用deque会超内存或者超时，要优化一下。