

# Assignment #3: March月考

Updated 1537 GMT+8 March 6, 2024

2024 spring, Compiled by 同学的姓名、院系

## 说明:

- 1) The complete process to learn DSA from scratch can be broken into 4 parts:
  - Learn about Time and Space complexities
  - Learn the basics of individual Data Structures
  - Learn the basics of Algorithms
  - Practice Problems on DSA
- 2) 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。
- 3) 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。
- 4) 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

### 02945: 拦截导弹

<http://cs101.openjudge.cn/practice/02945/>

思路:

dp

代码

```
1 '''
2 2100017810 刘思瑞
3 '''
4 num = int(input())
```

```

5 m = list(map(int,input().split()))
6 count = 1
7 dp = [0]*num
8 for i in range(num):
9     temp = [1]
10    for j in range(i):
11        if m[i] <= m[j] :
12            temp.append(dp[j]+1)
13            if dp[j] + 1 > count:
14                count = dp[j] + 1
15    dp[i] = max(temp)
16
17 print(max(dp))

```

代码运行截图、

## #44182421提交状态

---

状态: Accepted

源代码

```

'''
2100017810 刘思瑞
'''

num = int(input())
m = list(map(int,input().split()))
count = 1
dp = [0]*num
for i in range(num):
    temp = [1]
    for j in range(i):
        if m[i] <= m[j] :
            temp.append(dp[j]+1)
            if dp[j] + 1 > count:
                count = dp[j] + 1
    dp[i] = max(temp)

print(max(dp))

```

04147:汉诺塔问题(Tower of Hanoi)

<http://cs101.openjudge.cn/practice/04147>

思路:

## 代码

```
1 num,a,b,c = input().split()
2 num =int(num)
3 pin = [a,b,c]
4 def outputt(n,a,b):
5     global pin
6     print('%d:%s->%s' %(n,pin[a],pin[b]))
7 def secp(a,b):
8     m = [0,1,2]
9     m.remove(a)
10    m.remove(b)
11    return m[0]
12
13 def move(num,a,b):
14     if num == 0:
15         return
16     move(num-1,a,secp(a,b))
17     outputt(num,a,b)
18     move(num-1,secp(a,b),b)
19     return
20
21 move(num,0,2)
```

## 代码运行截图

## #44186939提交状态

状态: Accepted

源代码

```
'''
2100017810 刘思瑞
'''

num,a,b,c = input().split()
num =int(num)
pin = [a,b,c]
def outputt(n,a,b):
    global pin
    print('%d:%s->%s' % (n,pin[a],pin[b]))
def secp(a,b):
    m = [0,1,2]
    m.remove(a)
    m.remove(b)
    return m[0]

def move(num,a,b):
    if num == 0:
        return
    move(num-1,a,secp(a,b))
    outputt(num,a,b)
    move(num-1,secp(a,b),b)
    return

move(num,0,2)
```

### 03253: 约瑟夫问题No.2

<http://cs101.openjudge.cn/practice/03253>

思路:

链表

代码

```
1  '''
2  2100017810 刘思瑞
3  '''
4  class Node(object):
5      def __init__(self, item):
6          self.item = item
7          self.next = None
8
9  class ysf(object):
10     def __init__(self):
11         self.head = None
```

```

12     def is_empty(self):
13         if self.head:
14             return False
15         else:
16             return True
17     def append(self, item):
18         node = Node(item)
19         if self.is_empty():
20             self.head = node
21             node.next = self.head
22         else:
23             cur = self.head
24             while cur.next != self.head:
25                 cur = cur.next
26             cur.next = node
27             node.next = self.head
28
29 while True:
30     n,p,m = map(int,input().split())
31     if (n,p,m) == (0,0,0):
32         break
33     ysfi = ysf()
34     for i in range(1,n+1):
35         ysfi.append(i)
36     begin = ysfi.head
37     if p !=1:
38         for i in range(p-2):
39             begin = begin.next
40     else:
41         for i in range(n-1):
42             begin = begin.next
43     for j in range(n-1):
44         for i in range(m-1):
45             begin = begin.next
46         print(begin.next.item,end=',')
47         begin.next = begin.next.next
48     begin = begin.next
49     print(begin.item)

```

代码运行截图

状态: Accepted

源代码

```
'''
2100017810 刘思瑞
'''
class Node(object):
    def __init__(self, item):
        self.item = item
        self.next = None

class ysf(object):
    def __init__(self):
        self.head = None
    def is_empty(self):
        if self.head:
            return False
        else:
            return True
    def append(self, item):
        node = Node(item)
        if self.is_empty():
            self.head = node
            node.next = self.head
        else:
            cur = self.head
            while cur.next != self.head:
                cur = cur.next
            cur.next = node
            node.next = self.head

while True:
    n,p,m = map(int,input().split())
    if (n,p,m) == (0,0,0):
        break
    ysf1 = ysf()
    for i in range(1,n+1):
        ysf1.append(i)
    begin = ysf1.head
    if p !=1:
        for i in range(p-2):
```



21554:排队做实验 (greedy)v0.2

<http://cs101.openjudge.cn/practice/21554>

思路:

贪心

## 代码

```
1 n = int(input())
2 li = list(map(int,input().split()))
3 bi = [[li[i],i] for i in range(n)]
4 bi = sorted(bi,key = lambda x: x[0])
5 for i in bi:
6     print(i[1]+1,end=' ')
7 print()
8 summ = 0
9 for i in range(0,n):
10     summ+= (n-1-i)*bi[i][0]
11 print('%.2f' %(summ/n))
```

代码运行截图

状态: Accepted

源代码

```
n = int(input())
li = list(map(int,input().split()))
bi = [[li[i],i] for i in range(n)]
bi = sorted(bi,key = lambda x: x[0])
for i in bi:
    print(i[1]+1,end=' ')
print()
summ = 0
for i in range(0,n):
    summ+= (n-1-i)*bi[i][0]
print('%.2f' %(summ/n))
```

19963:买学区房

<http://cs101.openjudge.cn/practice/19963>

思路:

去年月考的题，就正常的操作题目

代码

```
1 summ = 0
2 n = int(input())
3 pairs = [i[1:-1] for i in input().split()]
4 distances = [sum(map(int,j.split(','))) for j in pairs]
5 value = list(map(int,input().split()))
```

```

6 vxjbb = value[:,:]
7 vxjbb.sort()
8 if n % 2 == 1:
9     midv = vxjbb[(n-1)//2]
10 else:
11     midv = (vxjbb[n//2]+vxjbb[n//2-1])/2
12 xjb = []
13 for i in range(n):
14     xjb.append(distances[i]/value[i])
15 xjbb = xjb[:,:]
16 xjbb.sort()
17 if n % 2 == 1:
18     midxjb = xjbb[(n-1)//2]
19 else:
20     midxjb = (xjbb[n//2]+xjbb[n//2-1])/2
21 for i in range(n):
22     if xjb[i]>midxjb and value[i] < midv:
23         summ+=1
24 print(summ)

```

代码运行截图

#42990980提交状态

[查看](#) [提交](#) [统计](#)

状态: Accepted

源代码

```

summ = 0
n = int(input())
pairs = [i[1:-1] for i in input().split()]
distances = [sum(map(int,j.split(','))) for j in pairs]
value = list(map(int,input().split()))
vxjbb = value[:,:]
vxjbb.sort()
if n % 2 == 1:
    midv = vxjbb[(n-1)//2]
else:
    midv = (vxjbb[n//2]+vxjbb[n//2-1])/2
xjb = []
for i in range(n):
    xjb.append(distances[i]/value[i])
xjbb = xjb[:,:]
xjbb.sort()
if n % 2 == 1:
    midxjb = xjbb[(n-1)//2]
else:
    midxjb = (xjbb[n//2]+xjbb[n//2-1])/2
for i in range(n):
    if xjb[i]>midxjb and value[i] < midv:
        summ+=1
print(summ)

```

基本信息

#: 42990980  
 题目: M19963  
 提交人: 23n2100017810  
 内存: 4260kB  
 时间: 23ms  
 语言: Python3  
 提交时间: 2023-12-07 16:07:54

27300: 模型整理

<http://cs101.openjudge.cn/practice/27300>

思路:

默认字典



## 代码

```
1  '''
2  2100017810 刘思瑞
3  '''
4  from collections import defaultdict
5  n = int(input())
6  d = defaultdict(list)
7  e = {'M':1,'B':1000}
8  for i in range(n):
9      name, attribute = input().split('-')
10     d[name].append(attribute)
11 kkk = d.keys()
12 kkk = list(kkk)
13 kkk.sort()
14 for i in kkk:
15     d[i].sort(key= lambda x: float(x[:-1])* e[x[-1:]])
16     print(i+":",end=' ')
17     for j in range(len(d[i])-1):
18         print(d[i][j],end=', ')
19     print(d[i][len(d[i])-1])
```

代码运行截图

状态: Accepted

源代码

```
'''
2100017810 刘思瑞
'''

from collections import defaultdict
n = int(input())
d = defaultdict(list)
e = {'M':1,'B':1000}
for i in range(n):
    name, attribute = input().split('-')
    d[name].append(attribute)
kkk = d.keys()
kkk = list(kkk)
kkk.sort()
for i in kkk:
    d[i].sort(key= lambda x: float(x[:-1])* e[x[-1:]])
    print(i+":",end=' ')
    for j in range(len(d[i])-1):
        print(d[i][j],end=', ')
    print(d[i][len(d[i])-1])
```

## 2. 学习总结和收获

大部分还是做过的题目，每日选做大概一周会做两三天题目

