

高三下信息算法综合练习3

1. 某批货物有 A、B 两个类别，每件货物都有对应价值。货物从 0 开始依次编号，并按编号从左往右升序放成一排。取走货物的规则如下：
- ①每次需要取走不同类且相邻的一对货物，若一排中有多对符合条件的货物，则最先取价值相差最小的一对货物（若价值相差相同取最左边的一对）。
 - ②一对货物取走后，该排货物的空白按原顺序补上。如原来该排货物编号依次为 0, 1, 2, 3, 货物 1, 2 取走后，该排货物编号变为 0, 3。
 - ③若该排货物中不再存在不同类且相邻的一对货物，则停止取货。
- 若货物数据如第 15 题图所示。

货物编号	0	1	2	3	4	5	6	7	8
货物类别	A	B	B	B	A	A	A	B	A
货物价值	7	8	6	9	10	3	1	5	8

第 15 题图

根据上述规则，能取走 4 对货物，依次为编号<0, 1>, <3, 4>, <2, 5>, <7, 8>。

- (1) 若将第 15 题图中 4 号货物变为 B，价值变为 6，则根据上述规则，最终剩余的货物为编号 ▲ 。（填数字）
- (2) 定义 insert_que(que, hwd) 函数，参数 hwd 由货物编号、相邻货物编号、货物价值差 3 个数据项组成，如 hwd 为 [3, 4, 1] 表示编号为 3 和 4 的货物价值差为 1。函数的功能是将可取的货物对 hwd 插入到列表 que 中，并按取货顺序依次排序。

```
def insert_que(que, hwd):  
    que.append(hwd)    #在列表 que 末尾添加一个元素 hwd  
    if len(que)!=1:  
        i=len(que)-2  
        while _____:  
            que[i]=que[i-1]  
            i=i-1  
        que[i+1]=hwd  
    return que
```

程序中加框处代码有错，请改正，并在划线处填入适合的代码。

- (3) 根据货物数据，模拟取货过程，计算取走货物的对数及顺序，实现上述功能的 Python 程序如下，请在划线处填入合适的代码。

```
def bl_link(que, link, head):    #寻找当前可取货物对  
    p=head  
    while link[p][1]!=-1:  
        q=link[p][1]  
        if _____ ① _____:  
            _____ ② _____  
            que=insert_que(que, [p, q, c])  
        p=link[p][1]  
    return que
```

’,’,

读取货物数据存入链表，其中货物编号即按货物数据顺序从 0 开始依次编号。列表 link 模拟链表结构，链表节点的前两个数据项依次为前驱指针和后继指针，第三个数据项为货物类别，第四个数据项为货物价值。链表中节点按货物编号升序排列，初始时 link[0] 为头节点，头节点指针 head 为 0。

例如有 4 件货物的类型依次为 A, B, B, A, 相应的价值为 7, 8, 6, 9, 相应的货物编号为 0, 1, 2, 3, 读取数据后，链表 link 为 [[-1, 1, 'A', 7], [0, 2, 'B', 8], [1, 3, 'B', 6], [2, -1, 'A', 9]], head=0。

代码略

’,’,

```
que=[];count=0
st=[False for i in range(len(link))]
que=bl_link(que, link, head)
while len(que)!=0:
    a=que.pop(0)      #将列表 que 第一个元素赋值给 a, 并将其从 que 中删除
    if not st[a[0]] and not st[a[1]]:
        st[a[0]]=True
        ③
        count+=1
        #存储取走的货物对编号，代码略
        p=link[a[0]][0]
        ④
        link[p][1]=q
        link[q][0]=p
        if ⑤:
            continue
        if link[p][2]!=link[q][2]:
            c=abs(link[p][3]-link[q][3])
            que=insert_que(que, [p, q, c])
#输出取走的货物对数 count 及依次取走的每对货物编号，代码略
```