

高三下信息选考算法综合练习 5

1. 对于任意正整数 a 和非负整数 b ，甲乙程序段输出结果相同，则乙程序段加框处的正确代码为 **C**

<pre>def f(a,b): if b == 0: return 1 res = f(a,b//2) if b%2 == 1: return res*res*a else: return res*res print(f(a,b))</pre>	<pre>res = 1 while b>0: if b%2==1: a = a * a b = b // 2 print(res)</pre>
---	---

- A. $res = res*res*a$ B. $res = res*res$ C. $res=res*a$ D. $res=a*a$

2. 定义如下函数：

```
def solve(a):
    stk = [0] * len(a)
    top = -1
    r = 255
    for i in range(len(a)-1, -1, -1):
        if a[i] > r:
            return False
        while top >=0 and a[i] <= stk[top]:
            r = stk[top]
            top -=1
        top +=1
        stk[top] = a[i]
    return True
```

执行语句 $res=solve(d)$ 后，若 res 的值为 True，则列表 d 不可能为 **B**

- A. [1, 4, 3, 2, 7, 6, 10, 9, 8] B. [3, 4, 1, 2, 5, 7, 8, 6, 9, 10]
 C. [1, 2, 4, 5, 3, 7, 8, 10, 9, 6] D. [1, 3, 2, 5, 6, 4, 8, 10, 9, 7]

[26年4月台州二模]3. 某工厂需采购一批材料，为实现总费用最低，需计划每日采购量。为了制定合理的每日采购量，工厂综合考虑材料的每日消耗量、市场每日的价格波动、库存量等因素，通过分析历史数据和市场趋势，先预测未来 n 天材料单价，再根据工厂每日固定消耗 x 单位的材料和仓库容量 v ($v \geq x$) 来模拟每天的采购量。

例如：预测未来 3 天该材料的单价分别为“5, 6, 4”（单位：元），现工厂每日消耗 1 个单位的材料，仓库容量为 5 个单位，则模拟采购计划量的过程如下：第 1 天（单价 5 元）：计划以当日单价采购 6 个单位材料，其中 1 个单位材料用于当日工厂生产，5 个单位材料临时存入仓库。第 2 天（单价 6 元）：当日单价高于仓库中的材料单价 5 元，消耗库存中 1 个单位的材料用于当日生产。再以当日 6 元单价计划采购 1 个单位材料临时存入仓库。第 3 天（单价 4 元）：当日单价低于仓库中单价 5 元和 6 元的材料，计划以当日单价采购 6 个单位材料，其中 1 个单位材料用于当日工厂生产，5 个单位材料用于替换仓库中原有的材料。3 天后，退还仓库中剩余的 5 个单位的材料，因此，材料的单价分别为“5, 6, 4”的计划采购量为“2, 0, 1”。

请回答以下问题(1) 若已知某工厂每日消耗 2 个单位的材料，仓库容量为 8 个单位，连续 4 天的原材料单价为“6, 4, 5, 3”（单位：元），计划采购量为 2, 4, 0, 2。

(2) 定义如下 use (a, head, x) 函数, a 链表每个元素包含三个数据项, 依次为第几天、剩余数量、指针, 链表已按材料单价升序, head 为头指针, x 为消耗量。函数功能是查找单价最低的材料用于消耗, 并返回剩余材料的头指针。

```
def use(a, head, x):
    y=x
    while head !=-1 and a[head][1] <= y:
        y -= a[head][1]
        head = a[head][2]
    a[head][1]-= y
    return head
```

回答下面两小题

①若 a=[[0, 5, -1], [1, 5, 2], [2, 1, -1]], head=1, x=1 执行这个函数, 返回值为 1。

②若函数里的语句 head=a[head][2]写成了 head=a[head][3], 在使用①中测试数据测试时, 程序一定会出现错误提示。这个说法对吗? B (填字母: A. 对 / B. 错)

(3) 编写 Python 程序如下, 请在划线处填入合适的代码。

"""天数、最大库存、每日固定消耗保存在变量 n、v、x, 每天对应的单价按时间顺序保存在列表 price 中, price[0]=1000 其中 1000 大于每天实际单价, price[i]为第 i 天原材料的单价 (i=1, 2, ..., n), 代码略"""

```
buy = [0]*len(price)
a = [[0, v, -1]]
head, buy[0] = 0, v
for i in range (1, len(price)):
```

```
    rep = 0
```

```
    q, p = -1, head
```

```
    while p != -1 and price[a[p][0]] <= price[i]
```

找插入位置

```
        q=p
```

```
        p= a[p][2]
```

```
    while p !=-1:
```

```
        rep += a[p][1]
```

buy[a[p][0]] -= a[p][1] p之后的节点价格都比刚插入的i天价格高, 那么后面这些节点每天的采购量都由i天来完成

```
        p= a[p][2]
```

```
    a.append ([i, rep+x, -1]) #为 a 追加一个元素
```

```
    buy[i] = rep+x
```

```
    if q !=-1: 如果插入位置不是链头
```

```
        a[q][2] = len(a)-1
```

```
    else: q== -1 插在链头
```

```
        head=len(a)-1
```

```
    head = use (a, head, x)
```

#退还仓库里剩余的材料, 代码略

```
total = 0
```

```
for i in range (1, len(price)):
```

```
    total += buy[i]*price[i]
```

```
print("第"+str(i)+"天的采购量为:", buy[i])
```

```
print("采购的最小费用为", total)
```

