

# 高三上信息练习卷（10月19日）

一、**选择题**（本大题共 12 小题，每小题 2 分，共 24 分。每小题列出的四个备选项中只有一个是符合题目要求的，不选、多选、错选均不得分）

阅读下列材料，回答第 1 至 3 题：

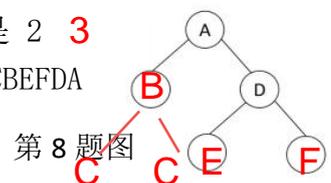
某校引入校园一卡通系统，师生员工可通过刷卡或刷脸完成生活消费、图书借阅、考勤签到等日常操作。所有数据均实时同步至后台数据库统一管理且可被食堂、财务处等部门使用。

1. 下列关于该系统中数据和信息的说法，正确的是 **D**
  - A. 该系统中存储的学生图像文件是 **信息** **数据**
  - B. 该系统中学生的数据仅存储在校园卡中
  - C. 数据库的应用降低了该系统的数据库管理效率
  - D. 数据可被多部门使用体现了信息的共享性
2. 下列关于该系统组成与功能的描述，**不**正确的是 **C**
  - A. 人脸验证过程主要由服务器完成
  - B. 系统的正常运行需软硬件协同工作
  - C. 系统的用户不包含数据库管理员
  - D. 分析学生就餐数据可优化食堂供餐策略
3. 为增强该系统信息安全性，下列做法**不**恰当的是 **C**
  - A. 为系统服务器设置防火墙
  - B. 定期对系统数据进行备份
  - C. 优化校园网络传输速度
  - D. 定期修改系统管理员密码

阅读下列材料，回答第 4 至 7 题：

无人机场系统由无人机场硬件（机库）、无人机等组成。机库所在位置的环境信息可由机库中配置的各类传感器获取。无人机通过内置的北斗定位、4G/5G 通信、视频及图像采集等模块实时采集飞行数据并上传服务器，同时借助摄像头与多种传感器持续获取环境数据，动态调整飞行姿态以实现智能避障。

4. 该系统中没有用到的技术可能是 **D**
  - A. 通信技术
  - B. 传感技术
  - C. 计算机技术
  - D. 数据可视化技术
5. 下列关于无人机实现智能避障的说法，**不**正确的是 **C**
  - A. 无人机实现智能避障属于领域人工智能
  - B. 增加训练数据的数量可提高智能避障的效果
  - C. 提升无人机摄像头的画质无法提高智能避障的效果
  - D. 仅依赖摄像头实现的智能避障无法确保成功
6. 下列关于无人机拍摄的图像和视频的说法，正确的是 **C**
  - A. 为减少图像传输时间，将 JPG 格式转换为 BMP 格式
  - B. 图像的容量与具体内容 **有关**
  - C. 图像数字化一般需要经过采样、量化与编码
  - D. 视频的帧频越高画面越清晰 **不一定**
7. 下列关于该系统中硬件和网络技术的说法，正确的是 **B**
  - A. 摄像头属于该系统的 **输出** 设备 **输入**
  - B. 无人机飞行数据可通过 5G 网络上传至服务器
  - C. 该系统的网络资源 **只有** 数据和软件 **硬件**
  - D. 无人机飞行数据传输至服务器 **无需** 遵循网络协议
8. 某二叉树的部分结构如第 8 题图所示，其前序遍历结果为 ABCDEF，下列说法正确的是 **D**
  - A. 该二叉树的深度可能是 4 **3**
  - B. 该二叉树的叶子节点数可能是 2 **3**
  - C. 该二叉树可能是完全二叉树 **不可能**
  - D. 该二叉树的后序遍历一定是 CBEFDA



入栈2次出栈第1次B, 入栈4次出栈2次D

9. 若元素入栈的顺序依次为 A, B, C, D, E, 约定操作: 当栈顶元素 P 出栈后, 出栈的总次数是已入栈次数的一半, 则将 P 元素出栈, 则最终栈中元素从栈底到栈顶的可能顺序为 **A**

- A. A C E                      B. A B D                      C. A C D                      D. A B E

10. 丑数是指仅包含质因子 2、3、5 的正整数, 且 1 被视为第一个丑数。编写如下甲、乙 Python 程序段实现判断正整数 n 是否为丑数, 下列对甲、乙程序段的描述, 不正确的是 **C**

<pre>def ugly(n):     for p in [2, 3, 5]:         while n % p == 0:             n = n // p     return n == 1</pre>	<pre>def ugly(n) :     if n == 1:         return True     for p in [2, 3, 5]:         if n % p == 0:             return ugly(n // p)     return False</pre>
甲程序段	乙程序段

- A. 甲、乙程序段分别使用了迭代算法和递归算法  
 B. 执行语句 f = ugly(25), 则甲程序段的函数返回结果为 True  
 C. 执行语句 f = ugly(12), 则乙程序段中函数 ugly 被调用的次数是 **3 次**    **4 次**  
 D. 对于同一个正整数 n, 甲、乙程序段加框处代码的执行次数不一定相同 **例如n=10,甲5次, 乙4次**
11. 有如下自定义函数:

```
def select(data, m):
    f = [0] * m ; ans = len(data) ; cnt = 0
    head = tail = 0 ; newdata = []
    while tail < len(data):
        f[data[tail]] += 1    投桶
        if f[data[tail]] == 1: cnt += 1    cnt记录当前窗口内不同数字的数量
        tail += 1
        while f[data[head]] > 1:    如果该数字第二次出现, 则出队
            f[data[head]] -= 1 ; head += 1
        if cnt == m and tail - head < ans:    cnt==m,即出现m个不同数字的无重复区域时, 且比当前最小窗口更小时, 更新ans
            ans = tail - head
            newdata = data[head:tail]
    return newdata
```

当参数 data 为 [1, 1, 3, 2, **2, 3, 0, 1,** 2, 0, 1, 3], m 为 4 时, 该函数的返回结果可能是 **B**

- A. [1, 1, 3, 2]              B. [2, 3, 0, 1]              C. [3, 0, 1, 2]              D. [2, 0, 1, 3]

12. 有如下 Python 程序段:

```
import random
a=[[3,05], [5,12], [2,2-1], [4,30], [1,43], [2,51]]
p = head = 4 ; q = a[p][1]
x = random.randint(1, 5)
while q != -1 and a[q][0] < x:
    p, q = q, a[q][1]
t = p
```

**1->4->3->2->5->2**

**链表遍历找第一个大于或等于x的节点**

高三技术学科 试题 第2页(共6页) **停止时:a[p][0]<x<=a[q][0]**

**↑  
7**

```

while q != -1:
    if a[q][0] < x:
        a[p][1] = a[q][1]
        a[q][1] = a[t][1]
        a[t][1] = q
        t, q = a[t][1], a[p][1]
    else:
        p = q ; q = a[q][1]

```

从前面的q位置继续往后遍历，如果有小于x的节点值，则删除后插入到t位置之后

1->4->3->2->5->2

#遍历链表 a ,依次输出 a[i][0] ,用空格间隔, 代码略

执行上述程序后, 下列输出结果不可能的是 **B**

- A. 1 4 3 2 5 2  $x=1$     B. 1 2 2 3 4 5    C. 1 3 2 2 4 5  $x=4$     D. 1 4 3 2 2 5  $x=5$

二、非选择题 (本大题共 3 小题, 其中第 13 小题 8 分, 第 14 小题 9 分, 第 15 小题 9 分, 共 26 分)

13. 某市搭建空气质量监测系统, 在市内各主要区域搭建气象站采集 PM2.5 浓度数据, 进行空气质量监测。对于每个传感器, 智能终端每小时获取 6 次数据, 计算 6 次数据中最大值和最小值的平均值作为该小时的平均浓度上传至服务器。服务器检测到异常情况时, 向管理员发送警示短信 (该功能在服务器端实现), 并通过智能终端控制喇叭发出警报。用户可通过手机 APP 或浏览器查看系统数据。请回答以下问题:

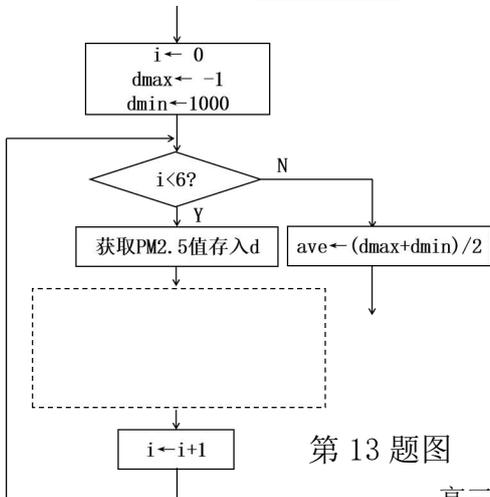
- (1) 下列选项中数据流向为双向的是 **B** (单选, 填字母: A. 智能终端和传感器/B. 服务器和浏览器)。  
 (2) 下列关于该系统智能终端的说法, 正确的是 **C** (单选, 填字母)。

- A. 智能终端**间接**控制执行器的打开或关闭 **直接**  
 B. 智能终端不具备数据处理功能  
 C. 智能终端和服务器的网络连接方式可采用有线或无线

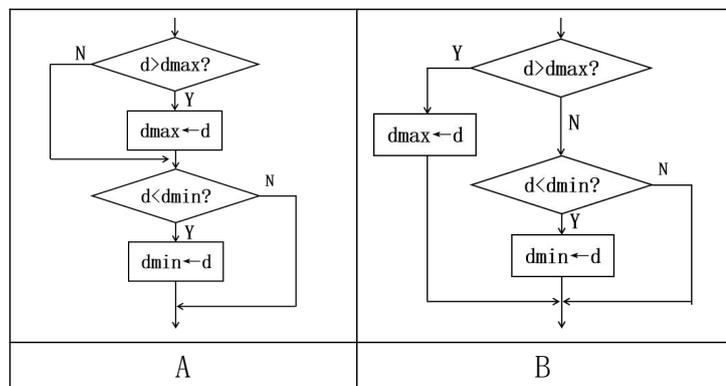
(3) 下列关于该系统支撑技术的说法, 正确的有 **ACD** (多选, 填字母)。(注: 全部选对的得 2 分, 选对但不全的得 1 分, 不选或有选错的得 0 分)

- A. 搭建该系统需要编写客户端程序 **手机app和浏览器说明C/S和B/S模式**  
 B. 传感器可以不通过智能终端直接将数据传输给服务器  
 C. 浏览器具备呈现服务器处理结果的功能  
 D. 系统搭建完成后, 在服务器上**运行程序**发现其中的错误属于**动态测试**

(4) 智能终端计算每小时 PM2.5 平均浓度的部分流程图如第 13 题图所示, 则图中虚线框处缺失的流程图可能是 **A** (单选, 填字母)。



第 13 题图



**B错原因:** 当 d 大于dmax 时, 程序只更新了最大值, 完全没去判断d 是否是最小值”, 导致最小值可能没有被正确更新。如果6个数据呈升序, 则dmin一直无法获取

A,服务器端控制发送喇叭示警指令给智能终端的代码有误, 导致无法发送指令  
 B,智能终端程序缺少响应服务器指令的程序模块(能接收无法响应)/智能终端上喇叭端口号设置错误/智能终端控制喇叭示警的程序有误

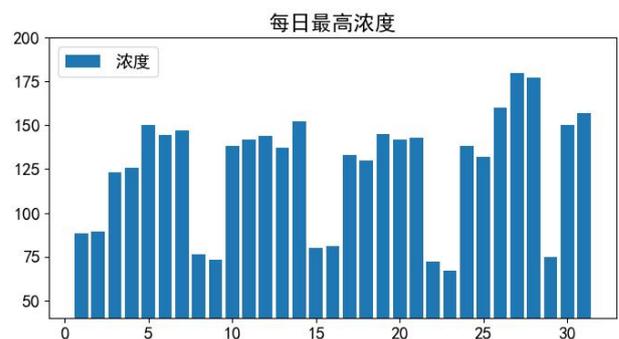
(5) 系统运行一段时间后, 管理员收到警示短信, 前往目标气象站后发现喇叭未发出警报。经调查, 数据库中数据与传感器采集数据一致, PM2.5 浓度超出正常范围, 且系统硬件设备和网络连接均正常。从程序角度分析可能是\_\_\_\_\_ (填字母: A. 服务器/B. 智能终端) 上的程序错误导致了该故障, 并描述一项可能导致该故障的具体原因。(根据选择的选项描述该对应的故障)

14. 空气质量监测系统已采集了该市一年的 PM2.5 浓度数据, 现要对这些数据进行分析, 请回答下列问题:

(1) 将气象站 1 的数据导出, 存储于 pmdata.xlsx 文件中, 如第 14 题图 a 所示。现要找出 PM2.5 月浓度均值最高的月份(数据只有一个), 并统计该月每日最高浓度的情况, 绘制如第 14 题图 b 所示的柱形图。实现上述功能的部分 Python 程序如下, 请选择合适的代码填入划线处(单选, 填字母)。

气象站	月	日	时	PM2.5
气象站 1	1	1	0	23.8
气象站 1	1	1	1	19.1
气象站 1	1	1	2	18.9
气象站 1	5	31	21	50.2
气象站 1	5	31	22	45.3
气象站 1	5	31	23	34.7

第 14 题图 a



第 14 题图 b

```
import pandas as pd
import matplotlib.pyplot as plt
df = pd.read_excel("pmdata.xlsx")
df1 = df.groupby("月", as_index = False).mean()
df2 = _____ ① D 排序, 根据最后一行判断是升序
#将 df2 最后一行的月份存入 m, 代码略
print("PM2.5 平均浓度最高的月份为: ", m)
_____ ② E 筛选最高月份的所有记录
_____ ③ A 根据“日”求该月每日浓度最高情况
plt.bar(df4["日"], df4["PM2.5"])
#设置绘图参数, 显示如第 14 题图 b 所示的柱形图, 代码略
程序中①②③处可选的代码有:
```

- A. df4 = df3.groupby("日", as\_index = False).max()
- B. df1.sort\_values("PM2.5", ascending = False)
- C. df4 = df3.groupby("PM2.5", as\_index = False).max()
- D. df1.sort\_values("PM2.5", ascending = True)
- E. df3 = df[df["月"] == m]
- F. df3 = df1[df["月"] == m]

(2) 将“气象站 1”中获取的 2024 年 1-12 月份的 PM2.5 浓度数据存储于列表 data 中(如 data=[["气象站 1", 3, 1, 0, 75.5], ["气象站 1", 10, 1, 1, 62.1], ..., ["气象站 1", 1, 31, 23, 40.6]]), 要求出各月份最大浓度和最小浓度的差值, 并输出最大差值(假设该值唯一)所在的月份。实现上述功能的部分 Python 程序如下, 请在划线处填入合适的代码。

```

#读取 PM2.5 浓度数据存入列表 data 中，代码略
#将 data 中的数据根据 PM2.5 浓度降序进行排序，代码略
n = len(data)
c = [[0, 0, False] for i in range(13)] 存放12个月的最高和最低浓度（0位为空）
maxm = 0
i = 0 ; j = n - 1
cnt=0 ①
while i < n and j >= 0 and cnt < 12:
    x = data[i][1]  data浓度降序排，x第一次获取每个月浓度最高值的月份，y第一次获取每个月浓度最高值的月份，x第二次碰到这个月时，值无效
    y = data[j][1]
    if c[x][0] == 0:
        c[x][0]=data[i][4] ② c[x]中存放x月的浓度最高值和最低值
    if c[y][1] == 0:
        c[y][1] = data[j][4]
    for k in [x, y]:
        if not c[k][2] and c[k][0] * c[k][1] > 0:
            if c[k][0]-c[k][1]>c[maxm][0]-c[maxm][1] ③:
                maxm = k
                c[k][2] = True
                cnt += 1
    i += 1
    j -= 1
print("差值最大的月份为：", maxm)

```

15. 已知由 M 个散件组装为 1 个完整件，用时需 T 分钟。某组装流水线工作流程为：散件按不同的批次汇聚至等待区。当等待区散件的个数达到 M 个时，组装设备启动，同时取走 M 个散件并进行组装；组装完成时，若等待区散件的个数达到 M 个，则直接进行下一轮组装。当“等待区散件的个数+某批次散件的个数≥阈值 TH”时，将该批次的散件整体进行转移加工。（每个批次的散件数量均小于 M；阈值 TH 大于 2×M）

编写程序模拟组装过程，计算组装完整件数量、剩余散件数量，并输出转移加工的情况。请回答下列问题：

(1) 各批次的散件数据已经按照送达时间顺序排列，若 M 为 5，则组装设备首次启动的时间为 **C** ▲ (单选，填字母：A. 6:00 / B. 6:15 / C. 6:18 / D. 6:29)

批次编号	送达时间	散件数量
LSH0001	6:00	2
LSH0002	6:15	2
LSH0003	6:18	3
LSH0004	6:29	1
...	...	...

到达M个

(2) 定义如下 cal(data)函数，将“送达时间”进行换算（如“9:03”换算结果为 543），其它数据值不变，程序加框处的代码有误，请改正。

```

def cal(data):
    res=[]
    for x in data:
        y=x[1].find(":")          # "Hello!".find("llo")结果为2
        h=int(x[1][:y])
        m=int(x[1][y+1:])
        res.append([x[0],h*60+m,x[2]])
    return res

```

(3) 实现其他功能的部分 Python 程序如下，请在划线处填入合适的代码。

```

def proc(data, T, M, TH):
    cnt=0 ;tmp=0;head=-1;tail=-1
    i=0
    n=len(data)
    curtime=0
    while i<n or tmp>=M:
        while i<n and data[i][1]<=curtime:
            if tmp+data[i][2]>=TH:
                data[i].append(-1)
                if head == -1:
                    head=i
                else:
                    data[tail][3]=i 或 data[tail][-1]=i
                    tail=i
            else:
                tmp+=data[i][2]
                i+=1
        if tmp>=M:
            tmp-=M
            curtime+=T
            cnt+=1
        elif i<n:
            tmp+=data[i][2]
            curtime=data[i][1]
            i+=1
    return cnt, tmp, head

```

链表代码

while停止时，i>=n或data[i][1]>curtime

凑到M个组装一次

没有凑到M个，但是还有批次的产品过来

此时data[i][1]>curtime，curtime跳过中间的空档

主程序部分：读入 M, T, TH; data 列表存放已按送达时间顺序排列的信息，每个元素包含批次编号、送达时间、散件数量 3 个数据项，代码略

```

data=cal(data)
c, t, p=proc(data, T, M, TH)
print("组装完整件共计",c,"个，剩余散件",t,"个。")
if p!=-1:
    print("所有批次的散件均无需转移加工。")
else:
    print("需转移加工的散件批次编号有：",end="")
    while p!=-1:
        print(data[p][0],end=" ")
        p=data[p][3]

```