

1	2	3	4	5	6	7	8	9	10	11	12
D	C	A	C	B	D	D	C	A	B	D	B

高三上信息综合练习(作业20)

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 2 题:

某校运动会采用智能管理系统。系统实时采集比赛数据(如短跑计时、跳远距离等),并自动生成排行榜。赛后,组委会将精选的比赛照片和视频上传至云平台,学生可通过学号登录系统查看个人成绩、下载照片。

1. 关于该系统中数据的说法,正确的是 **D**
- A. 短跑计时数据与照片的数字化过程完全相同
- B. 比赛数据无需经过加工和处理,就能生成排行榜
- C. 系统中的短跑计时、比赛照片和视频等数据均属于结构化数据
- D. 为提高该系统数据的管理效率,一般可将不同类型数据分别存储与管理

短跑计时数据(如电子计时器记录的时间)通常是直接由传感器或电子设备生成的数字信号(例如,起点和终点的红外光束被阻断的时间戳),其本身就是数字形式。

通常将结构化数据(如计时结果)存入数据库,非结构化数据(如照片、视频)单独存储(如文件系统)

2. 为保障系统安全并履行信息社会责任,下列做法合适的是 **C**
- A. 系统以明文存储与传输用户信息
- B. 组委会公开学生数据至社交媒体
- C. 系统管理员将密码设为复杂密码,并定期更新
- D. 学生将自己的账号借给他人,查询比赛成绩与照片

阅读下列材料,回答第 3 至 7 题:

某医院的智慧健康监测系统为患者配备了智能手环,内置心率传感器、血氧传感器、北斗定位和蓝牙模块,实时采集用户生理数据,并将数据加密后通过 5G 网络同步至云端服务器分析与存储。服务器端程序采用深度学习算法自动分析数据并在发现异常体征时触发预警;患者可通过手机 APP 查看健康报告,医生通过 NFC 刷卡快速调取病历,并生成可视化诊疗建议。

3. 在手机 APP 实时显示用户心率变化曲线的过程中,没有用到的技术是 **A**
- A. 人工智能
- B. 移动通信
- C. 数据可视化
- D. 传感器数据采集

4. 关于该系统中手机 APP 的功能与设计,说法正确的是 **C**

- A. APP 升级仅适配最新型号手环,还要兼容旧版
- B. APP 直接处理原始生理信号数据
- C. 在锁屏界面强制显示紧急异常预警通知
- D. APP 查看历史数据需从智能手环读取数据

APP 通常不直接处理原始生理信号,只需接收解析后的数据。
从服务器读取历史数据

5. 关于该系统的硬件与网络,说法正确的是 **B**

- A. 北斗定位系统参与传输患者的生理数据
- B. NFC 在此场景中的主要作用是近场身份识别
- C. 将患者数据加密传输,则无需网络协议的支持
- D. 发送预警信息主要体现了网络的分布处理功能

NFC(近场通信)技术常用于医疗场景中近场身份识别(如医生用手持设备刷患者手环快速获取身份信息),实现安全、便捷的数据关联(无需手动输入)。
数据传输功能

6. 系统需数字化患者皮肤照片用于 AI 分析。用扫描仪对 4cm×3cm 的皮肤区域采样,将其保存为分辨率为 800*600 像素,24 位真彩色的 BMP 位图。下列说法正确的是 **D**

- A. 该图像最多支持 24 种不同颜色
- B. 数字化后图像总像素点为 4×3=12 个
- C. 若皮肤深色斑点区域占比增大,文件存储容量会随之增加
- D. 将色彩深度调整为 256 色灰度,文件存储容量会减小为原来的 1/3

2²⁴色

800*600个

容量不变

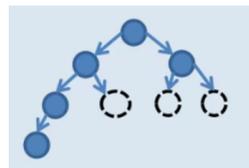
8位色

24位变8位

7. 若系统将健康数据误判为异常,最可能的原因是 **D**

- A. 5G 网络传输延迟
- B. 手环蓝牙连接断开
- C. 服务器硬件性能不足
- D. 深度学习的训练样本数据不足

判断错误一般是AI不够智能



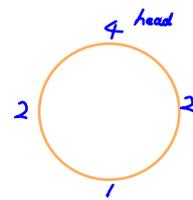
8. 某非完全二叉树包含 7 个节点,增加一个节点可将该二叉树变成完全二叉树,原二叉树的形态有多少种 **C**

- A. 1
- B. 2
- C. 3
- D. 4

9. n 个人排队买票,每人每次只能买一张票,并耗时 10 秒,买完后若还需买票,则重新到队尾排队(不计时间),无需买票则离开队伍。如:有 3 个人排队买票,每人买票数量依次为[4, 1, 2],则 20 秒后买票数据更新为[2, 3]。现有 4 个人排队买票,每人买票数量依次为[4, 2, 1, 2],则 30 秒后买票数据更新为 **A**

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

完全二叉树包括满二叉树,满二叉树是特殊类型的完全二叉树:当完全二叉树的最后一层也填满时,它就变成了满二叉树。



10. 有如下 Python 程序：

```
n = len(s)
max = c = 0
for i in range(n) :
    if "a" <= s[i] <= "z" or "A" <= s[i] <= "Z":
        c += 1
    else:
        if c > max:
            max = c
        c = 0
```

求最长单词个数，注意最后一个单词没有空格结尾，不计算

若 s 为 "Python is a popular language"，执行该程序段后，max 的值为 **B**

- A. 8
- B. 7
- C. 6
- D. 5

11. 有如下 Python 程序：

```
from random import randint
a = [9, 4, 7, 5, 6, 3, 8, 2]
n = len(a)
x = randint(1, 3)*2+1
for i in range(n-1) :
    for j in range(x, i, -1) :
```

[3,7]之间的奇数 3, 5, 7

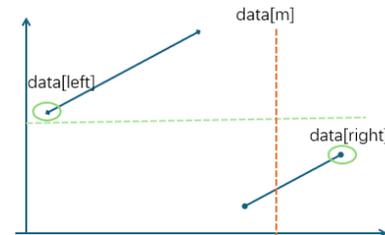
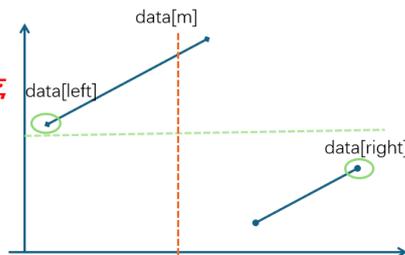
从x位开始向左冒泡：进行升序排序

执行该程序段后，数组 a 的值不可能为 **D**

- A. [2, 3, 4, 5, 6, 7, 8, 9] **x=7**
- B. [3, 4, 5, 6, 7, 9, 8, 2] **x=5**
- C. [4, 5, 7, 9, 6, 3, 8, 2] **x=3**
- D. [4, 5, 6, 7, 9, 3, 8, 2] **x=4**

12. 定义如下函数，返回 key 在列表 data 中的下标，若 key 不存在，则返回-1。

```
def bsearch(data, key) :
    left, right = 0, len(a) - 1
    while left <= right:
        mid = (left + right) // 2
        if data[mid] == key:
            return mid
        if data[left] < data[mid]:
            if data[left] <= key < data[mid]:
                right = mid - 1
            else:
                left = mid + 1
        else:
            if data[mid] < key <= data[right]:
                left = mid + 1
            else:
                right = mid - 1
    return -1
```



left~mid之间有序

轮转数子集

此时data[left]>=data[mid]

轮转数子集

如果调用函数返回结果不正确，则 data 可能是 **B**

- A. [32, 45, 48, 56, 59, 8, 12]
- B. [8, 59, 56, 48, 45, 32, 12] **不能分成两个有序子序列**
- C. [59, 8, 12, 32, 45, 48, 56]
- D. [8, 12, 32, 45, 48, 56, 59] **两个区间各自有序 且 data[0]>data[-1]**

二、非选择题（共 2 小题，其中第 13 小题 8 分，第 14 小题 9 分）

13. 某智慧农场监测大棚环境，智能终端通过传感器每小时采集 4 次土壤湿度数据，剔除最大值和最小值后取均值上传至服务器并存储到数据库中。服务器发现异常时向管理员告警，并通过智能终端向灌溉设备发送

控制指令。用户可通过手机 App 查看数据或远程控制灌溉设备。请回答下列问题：

(1)通过手机 App 向灌溉设备发送控制指令的正确流向顺序是 **A** (单选)。

- A. 手机 → 服务器 → 智能终端 → 灌溉设备
- B. 手机 → 智能终端 → 服务器 → 灌溉设备

常规思路：

$$\begin{aligned} &(\text{sum}(a, b, c, d) \\ &-\text{max}(a, b, c, d) \\ &-\text{min}(a, b, c, d)) / 2 \end{aligned}$$

(2)处理湿度数据过程中，剔除极值并计算平均值的操作在 **C** 完成(单选)。

- A. 服务器
- B. 灌溉设备
- C. 智能终端

(3)若数据库故障，可能引发的问题有 **AD** (多选)。(注：全部选对的得 2 分，选对但不全的得 1 分，不选或有错的得 0 分)

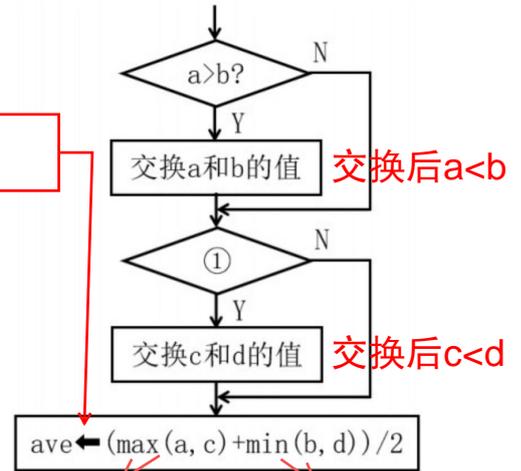
- A. 服务器无法保存湿度数据
- B. 灌溉设备无法执行启动指令
- C. 智能终端无法采集传感器数据
- D. 手机 App 无法查看历史湿度数据

注意流程图
赋值语句写法：

(4)将每小时获取的 4 个湿度数据分别存入 a、b、c 和 d 中，将剔除极值后的平均值存入 ave 的部分流程图如第 13 题图所示。图中

①处应填入 **c>d?** (注意，? 不能少)

(5)服务器检测到异常时需通过 Wi-Fi 向智能终端发送灌溉指令，用户手机 App 的灌溉指令也依赖网络传输。若大棚网络中断，系统将无法响应紧急灌溉需求，导致作物受损。请针对该缺陷设计一个改进措施。



表述要详细

去掉最小值 去掉最大值

在智能终端中添加检测到异常并控制灌溉设备的代码

智能终端额外添加 4g/5g 模块，通过 4g/5g 模块实现服务器与智能终端的数据通信

服务端与智能终端采用串口方式连接、备用网络

有线连接?

手动按钮? 数据通信?

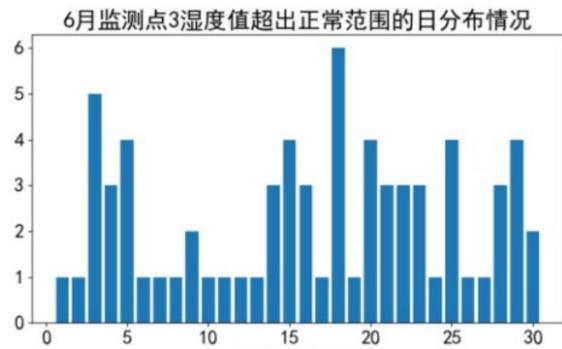
14. 土质监测系统已采集了大棚一年的土壤湿度数据，该大棚土壤湿

度的正常范围为 60-80(单位: %)。现要对这些数据进行分析，请回答下列问题：

(1)将监测系统的数据库导出，存于 humdata.xlsx 文件中，如第 14 题图 a 所示。现要找出 6 月份土壤湿度超出正常范围次数最多的监测点，并统计该监测点 6 月份土壤湿度超出正常范围的日分布情况，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)。

监测点	月	日	时	湿度
监测点1	1	1	0	80
监测点2	1	1	0	59
监测点3	1	1	0	71
监测点2	12	31	23	62
监测点3	12	31	23	75
监测点4	12	31	23	62

第 14 题图 a



第 14 题图 b

```
import pandas as pd
```

```
import matplotlib.pyplot as plt
```

```
df=pd.read_excel("humdata.xlsx")
```

```
df1=df[df.月==6]
```

```
df2=df1[(df1["湿度"]<60)|(df1["湿度"]>80)]#筛选 df1 中湿度值超出正常范围的数据
```

```
df_g=① (E) df2.groupby("监测点",as_index=False).count() #按监测点汇总次数
```

```
df_sort=② (D) df_g.sort_values("湿度",ascending=False) #找超出范围最多的监测点
```

```
#将 df_sort 首行的监测点存入 p, 代码略
```

```
df3=③ (B) df2[df2.监测点==p] #在超出范围的监测数据df2中筛选指定监测点的数据
```

```
df3_g=df3.groupby("日",as_index=False).count()
```

```
plt.bar(df3_g["日"],df3_g["湿度"])
```

#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：

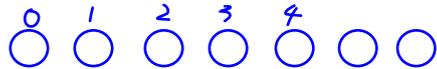
- A. df1[df1. 监测点==p]
- B. df2[df2. 监测点==p]
- C. df_g.sort_values("湿度", ascending=True)
- D. df_g.sort_values("湿度", ascending=False)
- E. df2.groupby("监测点", as_index=False).sum()
- F. df2.groupby("监测点", as_index=False).count()

(2)将 6 月某监测点的土壤湿度数据存储于列表 data 中，要求找出其中长度为 5 的连续子序列，使得该子序列的所有湿度值均在正常范围内，且其平均湿度最大(若存在多个，则选择最早出现的子序列)，并输出该子序列的平均湿度值及其起始下标(保证至少存在一个有效解)。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。

#读入土壤湿度数据，按采集的时间顺序存储于列表 data 中，代码略

max = -1

count=sum=0



for i in range(5): **先算前5个数据**

sum += ① data[i]

if not (60 <= data[i] <= 80): **不在正常范围 +1**

count +=1

if count == 0:

max = sum

start = 0

如果不存在超出范围的数据，当前5次连续湿度值均在范围内即找第一处符合要求的存在，作为max的初始值

i=1

while i<=len(data)-5:

sum -= data[i-1]

if not (60 <= data[i-1] <= 80):

count -= 1

② sum+=data[i+4] **出去一个i-1,进来一个i+4,使得i~i+4保存5个数长度**

if not(60 <= data[i+4] <= 80):

count += 1

if ③ count==0 and sum>max **#注意要满足两个条件：都在范围内（count==0），比最大值来的大**

max = sum

start = i

i+=1

print("最高平均湿度值："+str(max/5)+"，起始下标："+str(start))