

高三上信息选考练习卷14

第一部分 信息技术 (共 50 分)

一、选择题 (本大题共 12 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 1-2 题:

某网络学习资源平台支持用户上传文字、图片、音频、视频等原创学习资源到服务器并为个人资源设置访问权限, 在个人中心显示个人资源被访问与下载的相关数据。

1. 下列关于数据的说法, 正确的是 **D**

- A. 文字、图片学习资源数据属于结构化数据
- B. 为便于传输, 将 JPEG 格式的图片转为 BMP 格式上传
- C. 可利用个人资源被访问数据分析该用户的学习偏好
- D. 同一个学习资源对不同用户而言价值是不一样的

被访数据分析的是访问者的偏好, 而不是当前用户的偏好

2. 下列行为中, 符合信息安全与信息社会责任的是 **B**

- A. 将服务器中的学习资源以明文形式存储
- B. 平台管理员定期备份服务器中的学习资源
- C. 未经平台用户同意在网络售卖其分享的资源
- D. 使用他人账户在平台上传学习资源

阅读下列材料, 回答第 3-6 题:

某高中智慧校园系统中各教室的电子班牌内置 4G/5G 通信、音频和温湿度采集、学生上课行为识别等模块, 实时采集、处理教室各项数据, 然后传输至服务器; 服务器分析学生行为数据, 基于系统内数十万份案例, 为学生提供可选择的提高学习效率方案; 教师可通过智慧校园 app 或在浏览器上登录智慧校园网站查询班级各项数据。

3. 下列关于该系统中硬件和软件的说法, 正确的是 **A**

- A. 音频采集模块能将模拟信号转换为数字信号
- B. 音频与温湿度数据的采集时间间隔不能相同
- C. 系统处理数据的程序只能部署在服务器端
- D. 智慧校园 app 属于教师移动终端的系统软件

智能终端也可以

4. 下列技术中, 不能用于电子班牌向服务器传输数据的是 **C**

- A. 4G/5G
- B. Wi-Fi
- C. NFC
- D. 光纤通信

5. 为学生提供高效学习方案是通过基于规则学习的符号主义人工智能方法来实现的, 下列说法不正确的是 **C**

- A. 通过规则学习提供方案需要庞大的知识库
- B. 系统提供的方案可能完全不适合该学生
- C. 基于规则学习来提供方案属于强化学习范畴
- D. 不断更新系统内的案例能提高推荐方案的准确度

符号主义

行为主义

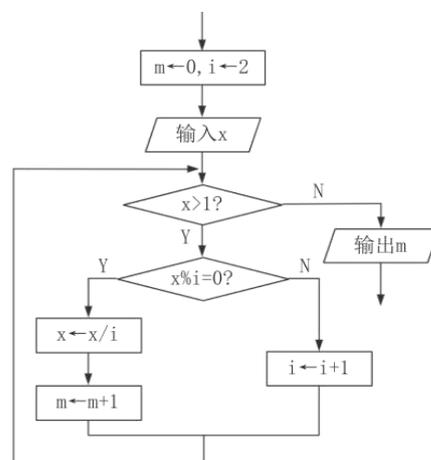
6. 该高中共有 60 个班, 电子班牌每隔半小时采集温湿度数据上传到服务器, 一天共有 48 个时间点, 若使用二进制对实时传输到服务器的温湿度数据进行编码, 前几位表示班级, 其余位表示采集时间点, 则所需的二进制位数最少为 **B**

- A. 14
- B. 12
- C. 10
- D. 8

7. 某算法的部分流程图如第 7 题图所示, 执行这部分流程, 若输入 x 的值为 120, 则输出 m 的值是 **C**

- A. 3
- B. 4
- C. 5
- D. 6

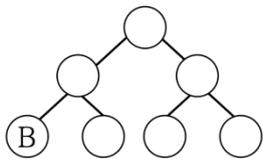
m 是质因子的个数



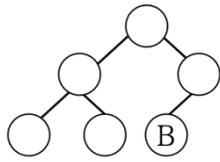
第 7 题图

C在右子树

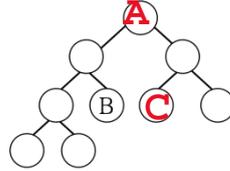
8. 节点 A 是某完全二叉树（层数>2）的根节点，节点 B 是该二叉树度为 0 的节点，节点 C 是节点 A 的右孩子节点。在中序遍历序列中节点 B 位于节点 A 的左侧区域，在前序遍历序列中节点 B 与节点 C 相邻。下列二叉树，节点 B 位置标注正确的是 **C** **B一定在左子树** **左子树的最左边B和右子树的最右边C**



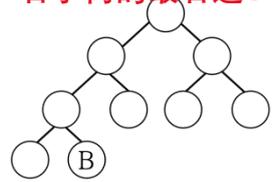
A.



B.



C.



D.

9. 栈 k 初始为空，已知元素入栈顺序为 $\overset{e}{2}, \overset{a}{7}, \overset{b}{3}, \overset{f}{1}, \overset{6}{6}, \overset{d}{9}$ ，元素出栈顺序为 $\overset{a}{a}, \overset{b}{b}, \overset{c}{c}, \overset{d}{d}, \overset{e}{e}, \overset{f}{f}$ ，若 $b=6, e=2$ ，则 f 的值是 **D**

A. 7

B. 3

C. 1

D. 9

10. 有如下 Python 程序段：

```
s="python"; head=tail=0; q=[""]*10
for i in s:
```

```
    if head!=tail and i>q[head]:
```

```
        q[tail]=q[head]
```

```
        head+=1
```

```
    else:
```

```
        q[tail]=i
```

```
        tail+=1
```

```
print(q[head:tail])
```

执行程序段后，输出的结果是 **B**

A. ['h', 'p']

B. ['p', 'h', 'o', 'n']

C. ['h', 'p', 'n']

D. ['p', 'h']

11. 有如下 Python 程序段：

```
a=[8,2,4,3,1,4,4,7,1,4]; b=[8,4,1,3,6,5,9,2,7,0]
```

```
i,j=0,len(a)-1; key=int(input("输入待查找的数:")); s=""
```

```
while i<=j:
```

```
    m=(i+j+1)//2
```

```
    if a[b[m]]>key:
```

```
        j=m-1
```

```
        s+="L"
```

```
    else:
```

```
        i=m+1
```

```
        s+="R"
```

```
print(s)
```

执行该程序段后，输入 key 的值为 4，输出结果是 **A**

A. RLR

B. RLLR

C. LRLR

D. LRL

12. 有如下 Python 程序段：

```
import random
```

```
a=[0]*5
```

```
for i in range(5):
```

```
    a[i]=random.randint(1,4)*2
```

```
for i in range(4):
```

```
    if a[i]>a[i+1]:
```

```
        a[i],a[i+1]=a[i+1],a[i]
```

```
    else: a[i]<=a[i+1]时
```

```
        a[i]-=1 使得a[i]一定小于a[i+1]
```

```
print(a)
```

一趟循环，只能把最大数往后送

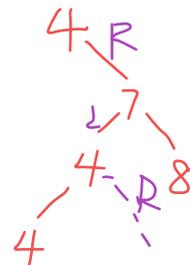
索引数组，升序排序

a[b[0]]<a[b[1]].....

0 1 2 3 4 5 6 7 8 9

1, 1, 2, 3, 4, 4, 4, 4, 7, 8

相等时往后找



执行该程序段后，输出的结果不可能是 **B**
 原：[8,2,4,6,6] 第一个8会变成7 [4,2,6,8,8] [2,4,6,6,8]
 A. [2,4,6,6,8] B. [2,4,6,8,8] C. [2,3,6,7,8] D. [1,4,5,6,8]

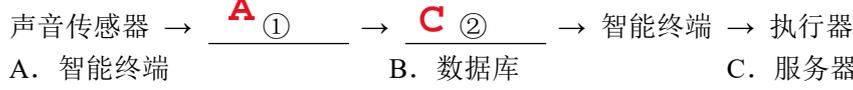
二、非选择题（本大题共3题，其中第13小题8分，第14小题9分，第15小题9分，共26分）

13. 学校兴趣小组搭建噪声监测系统，在某小区人流量大的 **5个区域设置监测点**，智能终端每隔10分钟获取1次声音传感器采集的声音数据，通过网络将声音数据传输到服务器。服务器根据数据判断为噪声时，通过智能终端控制执行器发出预警信号。小组成员通过浏览器可查看历史监测数据。请回答下列问题：

(1) 搭建该监测系统时，声音传感器与智能终端的配备总数量合理的是 **A**（单选，填字母：A. 5个声音传感器和5个智能终端 / B. 5个声音传感器和1个智能终端）。

(2) 下列功能只能在服务器端程序中实现的是 **C**。（单选，填字母：A. 控制执行器发出预警信号 / B. 调整声音数据的采集时间间隔 / C. 响应浏览器的访问请求）

(3) 系统采集声音数据到发出预警信号过程中的数据流向如下所示，请选择合适的硬件填入划线处（填字母）。



(4) 兴趣小组对采集到的声音数据进行分级：声音值不超过50，为一级；声音值超过90，为三级；其余声音值为二级。下列Python程序段中符合要求的有 **BC**（多选，填字母）。（注：全部选对的得2分，选对但不全的得1分，不选或有错的得0分）

A. <pre>t="一级" if v>50: t="二级" else: t="三级"</pre>	B. <pre>if v>50: t="二级" if v>90: t="三级" else: t="一级"</pre>	C. <pre>if v>90: t="三级" elif v>50: t="二级" else: t="一级"</pre>	D. <pre>if v<=50: t="一级" if v>90: t="三级" else: t="二级"</pre>
--	--	--	---

只有两种情况 **会覆盖一级**

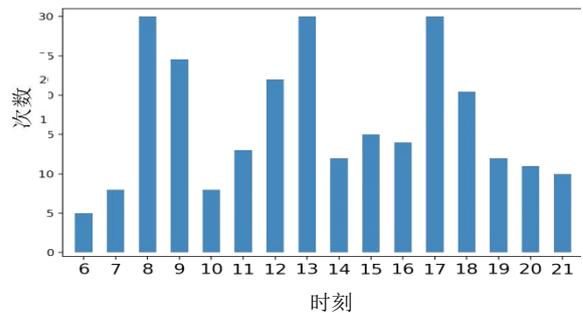
(5) 兴趣小组用浏览器查看监测页面，页面动态显示5个区域最新的声音数据及其采集时间。系统正常工作一段时间后，兴趣小组在监测页面发现某一区域的监测数据不再变化，其余区域正常动态显示，隔1小时后刷新，该区域监测数据仍不变。简要说明系统中可能造成上述问题的原因 **该区域的IoT损坏、智能终端损坏；该区域IoT与智能终端连接异常**。（注：回答2项，1项正确得1分）

14. 噪声监测系统已经采集该小区9月份的声音数据，该小区规定声音值>50为噪声，系统监测到噪声时会自动发出预警信号。现要对这些数据进行分析，请回答下列问题：

(1) 将监测点1的数据导出，存储在 data.xlsx 文件中，部分数据如第14题图 a 所示。若某天中某一时刻的预警次数超过4次，则该时刻被判定为当天的“高频预警时刻”。现统计高频预警时刻的分布情况并绘制如第14题图 b 所示的柱形图。实现上述功能的部分 Python 程序如下。

监测点	日	时	分	声音值
监测点1	1	0	0	37
监测点1	1	0	10	39
监测点1	1	0	20	25
监测点1	30	23	30	24
监测点1	30	23	40	32
监测点1	30	23	50	36

第14题图 a



第14题图 b

```

import pandas as pd
import matplotlib.pyplot as plt
df=pd.read_excel("data.xlsx")
df.insert(5,"标记",0) #插入“标记”列，值为0
for i in df.index:
    if df.at[i,"声音值"]>50: #将噪声数据标记为1
        df.at[i,"标记"]=1 或 df["标记"][i]=1
d=[0]*24
for i in range(1,31): 天数
    
    for j in df1.时:
        d[j]+=1

```

#筛选出列表 d 中值不为 0 的数据，绘制如第 14 图 b 所示的 高频预警时刻 分布柱形图，代码略。

①请在划线处填入合适的代码。

②方框中应填入的语句依次为 **AFD** (选 3 项，填数字序列，少选、多选、错选或次序错均不得分)。

- A. df1=df[df.日==i]
- B. df1=df[df.时==i]
- C. df1=df1[df1.声音值>4]
- D. df1=df1[df1.标记>4]
- E. df1=df1.groupby("时",as_index=False).count()
- F. df1=df1.groupby("时",as_index=False).sum()

统计“高频预警时刻”。for 循环 i 取值从 1 到 30 可分析出 i 的含义为“日”，故 AB 中选出 A，分别对每一天数据进行处理。再按照“时”列进行分组处理，为了知晓每个“时刻”预警了几次来定位“高频时刻”，所以分组后进行 sum() 计算，此时“标记”列累加的值就是该时刻报警次数。不能用 count(因为不管“标记”列是 0 还是 1 都会被 count() 算在内)。

(2) 在第 1 小题的基础上，计算监测点 1 出现次数最多的高频预警时刻，若有多个则一并输出。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。

```

t=[i for i in range(24)]
i=0
while i<len(d): len(d)-1,i,-1
    for j in range(①):
        if d[j]>d[j-1]:
            d[j],d[j-1]=d[j-1],d[j]
            t[j],t[j-1]=t[j-1],t[j]
        if i!=0 and d[i]!=d[i-1]:
            break
    i+=1
    t[0:i]或 t[:i]
print("监测点 1 噪音预警次数最多的时刻: ",②)

```

用冒泡优化程序实现找最大的连续若干个值，外层 while 循环控制冒泡的轮数，当某一轮找到的最大值 d[i] 和前面值 d[i-1] 不同时直接结束排序过程，所以内层 for 循环实现从右往左冒。根据 i 初始值为 0 和冒泡比较的是 d[i] 和 d[i-1] 可知 ① 空为 len(d)-1,i,-1 或 len(t)-1,i,-1 或 23.i,-1。根据 break 退出排序的条件可知从索引 0 到 i-1 是找到的连续最大的值，要输出的是预警次数最多的时刻，t[0:i] 或者 t[:i]。

(3) 请结合第 2 小题的程序代码，观察第 14 题图 b 所示的柱形图，则监测点 1 噪音预警次数最多的 3 个时刻为 ▲ (请用逗号间隔)。

8,13,17

15. 学校某学科竞赛训练平台共有“ABCDE”5个模块供学生日常训练。平台根据学生训练数据给学生标记类型，例如该生在“AC”两个模块错误率高，则该生的类型为“AC”。

平台为学生生成个性化训练题单，生成规则：先按编号从小到大遍历未生成题单的学生，依次从同类型学生的训练数据错误率前5题中，按错误率从高到低挑选该生未做过的题目加入题单，直到满足题单设置的题量（在遍历过程中选题是相互的）；若无法满足，再遍历已生成题单的同类型学生，按相同规则挑选题目加入题单；若仍无法满足，最后自动从题库中选题补足。

部分学生数据如第15题图所示：

编号	姓名	类型	训练数据（已练题号，按错误率降序排列）
1	学生1	B	16,8,2,9,3,7,10
2	学生2	ABD	5,9,2,4,11,3,21,17
3	学生3	B	9,1,4,12,15,8,16
4	学生4	ABD	1,7,3,13,4,9,11
5	学生5	ABD	20,4,9,12,3,6,1
...

第15题图

如设定题单题量为3，为“学生1”生成题单的过程如下：根据规则，“学生1”遍历到1个类型相同者“学生3”，因此“学生1”从“学生3”训练数据的前5题中依次挑选3道未做过的题目，依次为“1,4,12”。所以“学生1”的题单为“1,4,12”；同时“学生3”从“学生1”训练数据的前5题中依次挑选3道未做过的题目，依次为“2,3”，由于未达到设定题量，“学生3”的题单还需继续生成。

编写程序模拟生成学生个性化训练题单过程，请回答下列问题：1, 7, 13, 20, 12

- 若学生数据如第15题图所示，题单题量为5，则“学生2”的题单为 ▲ 。（题号之间用逗号间隔）
- 定义如下 `select(a,b,c,d)` 函数，参数 `a,b,c,d` 分别对应题单主人已练题号、类型相同者已练题号、题单题号和题单设定的题量。函数功能是将 `b` 中题号按照规则分配给该生题单。

```
def select(a,b,c,d):
```

```
    for i in range(5): #遍历类型相同者已练题号前5题
```

```
        if not(b[i] in a) and not(b[i] in c) and len(c)!=d:
```

```
            c.append(b[i])
```

```
    return c
```

- 1、同类型学生训练数据错误率前5题；
- 2、挑选该生未做过的题目 `not(b[i] in a)`；
- 3、该题目为未入选题单 `not(b[i] in c)`；
- 4、总题数不超过 `len(c)!=d`。
删除条件 `len(c)!=d` 可能导致题单数目超过设定值

若函数 `select` 中加框处代码误写为“`not(b[i] in a) and not(b[i] in c)`”，调用函数后，该错误可能导致的结果是 D▲（单选，填字母）。

- A. 题单中出现题单主人已练过的题号
- B. 题单中出现非题单主人薄弱模块的题号
- C. 题单的题量比预设题量少
- D. 题单的题量比预设题量多

(3) 实现模拟生成学生个性化题单的程序如下，请在划线处填入合适的代码。

```
"""
读取学生数据，存入列表 data，列表每个元素包含 4 个数据项，分别为编号、姓名、类型、已
练题号，
列表 data 中的数据已按编号升序排列，代码略。
例如 data=[[1,"学生 1","B",[19,8,2,9,3,7,10]], [2,"学生 2","ABD",[5,9,2,4,11,3,21,17]], .....]。
"""
num=int(input("请输入题单的题目数量： "))
n=len(data)
f=[0]*n
for i in range(n):
    data[i].append([]) #利用 append 函数向学生数据中追加空列表存储题单题目编号
    data[i].append(-1)
dic={}
i=0
while i<n: and f[i]==0
    f[i]=1
    weak=data[i][2]
    if weak in dic:
        data[i][5]=dic[weak]
        dic[weak]=i
    j=i+1
    while j<n:
        if data[j][2]==weak and f[j]==0
            ② data[j][2]==data[i][2] and f[j]==0
            data[i][4]=select(data[i][3],data[j][3],data[i][4],num)
            data[j][4]=select(data[j][3],data[i][3],data[j][4],num)
            if len(data[j][4])>=num:
                data[j][5]=dic[weak]
                dic[weak]=j
                f[j]=1
            if len(data[i][4])>=num:
                break
        j=j+1
    while k!=-1 and len(data[i][4])<num:
        data[i][4]=select(data[i][3],data[k][3],data[i][4],num)
        k=data[k][5]
    if len(data[i][4])<num: #遍历所有相同类型者后题单题目数量仍不足
        #从题库分配新题给该生，代码略。
    j=i+1
#输出每个学生的个性化训练题单，代码略
```

头插，dic 存放链头指针

dic[weak]=i

先在未生成题单的学生中找

data[j][2]==weak and f[j]==0

② data[j][2]==data[i][2] and f[j]==0

双向判断，将后面可以生成题单的都插入到链表中

如果当前 i 可以生成题单，就不再找了

如果 i 在未生成题单的人中找不到，则返回到已经生成题单的链表中继续找
k=data[i][5] 或 k=dic[weak] 或者 k=i