



技术参考答案

第一部分：信息技术（共 50 分）

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、错选、多选均不得分）

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

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. (1) B 1 分

(2) C 1 分

(3) AC 2 分（漏选给 1 分，错选、多选不得分）

(4) $ave = (s - \max x - \min x) / 58$ 2 分

(5) 通过编码器设置报警器的工作模式，例如只亮光不发出声音判断为 CO₂、只发出声音不亮光为 PM_{2.5}，既发光又发声表示两种数据
或使用多色 LED 灯珠控制灯的颜色来区别
或设置报警器的音调来进行区别
或定义语音报警

这类有点对报警器编码之类的答案，都可以给分……2 分

14. (1) A 1 分

(2) ①A 1 分

②C 1 分

(3) ① $t[14:16]$ 或 $t[14:]$ 2 分

② $abs(minute(df.at[i+1, "打卡时间"]) - minute(df.at[i, "打卡时间"])) \geq 120$
不加 abs 也给分 2 分（没有 $minute$ 函数或没有 $df.at[]$ 扣 1 分）

③ $cnt = 0$ 2 分

15. (1) 14 1 分

(2) ① $que[k-1][1] = head$ 2 分

②C 2 分

(3) ① $que[d-1][0] = data[temp[-1]][4]$

或 $que[d-1][0] = data[temp[len(temp)-1]][4]$ 2 分

② $d = unit[0]$ 2 分

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

高三上技术选考练习卷2 信息技术部分

一、选择题（本大题共12小题，每小题2分，共24分。）

阅读下列材料，回答第1至4题。

某博物馆推出线上预约功能，游客预约后刷身份证即可通过入口闸机入馆。馆内设有AI数字人导览员，游客可通过语音对话与其实时互动，获取个性化导览及展品讲解服务；也可扫描部分展品旁的二维码或通过自助语音导览设备自动播放讲解，其中自助语音导览设备搭载RFID技术，能精准识别所观展品的电子标签。

1. 下列关于数据的说法，正确的是 **D**

- A. 游客预约的数据存储在入口闸机中 **服务器中**
- B. 某藏品为刻有甲骨文的龟甲，甲骨文通过龟甲来记录体现了信息的时效性 **载体依附性**
- C. 游客可进行拍照，其过程实现了从数字信号到模拟信号的转换 **模数转换**
- D. 游客扫描二维码收听展品讲解的过程，体现了从数据中获取信息的过程 ✓

2. 下列措施中，关于数据安全的做法不合适的是 **C**

- A. 对游客信息进行加密处理
- B. 定期对服务器数据进行备份
- C. 为所有员工提供相同权限
- D. 为系统服务器配备UPS（不间断电源）

3. AI数字人导览员识别游客语音的过程是 **基于神经网络** 方法实现的，下列说法不正确的是 **D**

- A. 该方法不用事先定义语音中“‘藏品’该怎么念”等知识，而是直接从数据出发
- B. AI数字人导览员识别游客语音的结果并不总是正确的
- C. 该识别功能使用了数据驱动的人工智能方法
- D. 在该识别的学习方法中需要不断尝试各种解决问题的可能途径 **试错：行为主义**

**联结主义
神经网络
数据驱动
数据训练
深度学习**

4. 馆内展品电子标签本身 **具备电源**，下列说法正确的是 **A**

- A. 电子标签能够主动向自助语音导览设备发送射频信号
- B. 通过RFID能够定位展品，说明信息系统利用传感技术实现对外部世界的控制
- C. 自助语音导览设备属于RFID读写器，发射的射频不具备远距离传播能力
- D. 该RFID需要在电子标签与自助语音导览设备之间建立机械接触

**类似etc设备，
有源电子标签，
主动发射信号，
通信距离相对
无源距离远**

阅读下列材料，回答第5至6题。

某学校启用了智慧餐厅系统，功能如下：学生自助选取餐品并刷卡支付；电子秤实时采集、存储餐品信息，自动计算价格与热量，同时将数据上传至服务器。教师和家长可通过移动终端的专用APP，随时查看学生的饮食情况。

5. 下列关于该系统的组成与功能说法正确的是 **C**

- A. 该信息系统的用户由学生、教师和家长组成
- B. 该系统可以共享的资源只有专用APP和饮食数据 **共享的资源有：硬件资源、软件资源、数据资源**
- C. 系统可以智能推荐菜品，说明该系统具有数据挖掘与分析的功能
- D. 电子秤中不一定有处理器部件

6. 下列关于该系统中网络技术的说法正确的是 **A**

- A. 若移动终端与服务器属于同一个局域网，则无需经过网关 **处于不同局域网，则需要路由器（网关）**
- B. 支付数据传输至服务器的过程无需网络协议的支持 **网络传输时 必须遵照协议**
- C. 服务器一般使用动态地址，可降低数据上传的安全性风险 **服务器IP地址一般固定（静态IP地址）**
- D. 若采用无线网络传输数据，则无需传输介质 **客户端IP地址可动态**

7. 某 **完全** 二叉树的中序遍历为 **CBDEA** ，其中A为根节点，则方框中节点数量的可能情况有 **A**

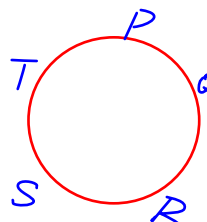
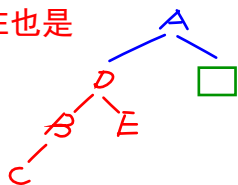
- A. 1种
- B. 2种
- C. 3种
- D. 4种

8. 某队列初始为：P，Q，R，S，T（队首到队尾）。元素出队后直接输出或重新入队。已知最终输出序列是 Q，S，P，T，R，且在整个过程中，元素P重新入队的次数不能超过1次。满足以上条件时，元素R的最少重新入队次数是 **B**

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

在完全二叉树中，任意节点的子树仍然是完全二叉树。

左子树CDBE也是完全二叉树



9. 有如下 Python 程序段: **碰到数字字符k, i往后跳1+k个**

```
s = "a2b4c3d1e"
i, r = 0, ""
while i < len(s):
    r += s[i]
    step = 1
    if "0" <= s[i] <= "9":
        step += int(s[i])
    i += step
```

运行该程序段后, r 的值为 **B**

- A. "a2b4c3" B. "a2c3"✓ C. "a2c3e" D. "a2b4c3d1"

10. 有如下 Python 程序段:

```
def f(n):
    if n <= 2:
        return n
    else:
        return f(n-1) + 2 * f(n-2)
result = 0
for i in range(1, 6):
    result += f(i)
```

程序运行后 result 的值是 **A**

- A. 31 B. 32 C. 63 D. 64

从f(1)逆向开始写:

$f(1)=1$
 $f(2)=2$
 $f(3)=f(2)+2*f(1)=2+2*1=4$
 $f(4)=f(3)+2*f(2)=4+2*2=8$
 $f(5)=f(4)+2*f(3)=8+2*4=16$

$f(1)+f(2)+f(3)+f(4)+f(5)=31$

11. 有如下 Python 程序段:

```
n = len(data)
mark = [False] * n; res = [0] * n
for i in range(n):
```

if not mark[i]: **节点未访问过**

count = 0; p = i

while p != -1 and not mark[p]: **节点未访问**

mark[p] = True **走过的节点标记为True**

count += 1 **从i开始, 没有访问过的节点个数**

p = data[p][1]

q = i **从i开始重新走**

mark: [True, True, True, True, True]

while q != -1 and res[q] == 0: **res: [4, 4, 4, 4, 1]**

res[q] = count

q = data[q][1]

H

若 data 为 [[36, 1], [72, 3], [15, -1], [90, 2], [28, 0]], 程序运行后, 下列说法正确的是 **D**

A. $res[1] + res[3] = 4 + 4 = 8$ B. mark 中存在某个元素的值为 False **×**

C. 列表 res 中, res[4] 的值最大 **×**

D. 存在某个索引 i, 使得 res[i] 的值等于从 i 出发 (含 i) 沿链表到尾节点的节点总数

12. 已知一个长度为 n (n>=3) 的整数数组 a 满足以下条件: 数组元素大小呈先升后降, 仅存在一个索引 i (0 < i < n-1) 使得 $a[i-1] < a[i]$, 且 $a[i] > a[i+1]$ 。这样的数组被称为山脉数组。定义以下函数实现在山脉数组中找到峰值元素 (即最大值) 的索引。

```
def search(a):
```

left, right = 0, len(a) - 1

while left < right: **退出循环时 left==right**

mid = (left + right) // 2

if ①:

②

else:

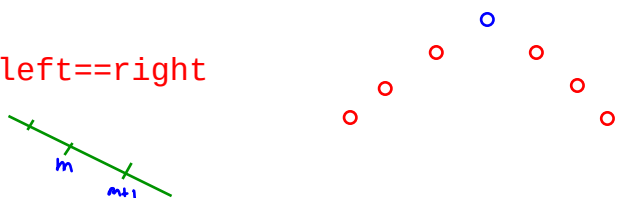
③

return left, right **相同**

划线①②③处应填入的代码为: **B**

- A. ① $a[mid] > a[mid + 1]$
 ② $right = mid + 1$ **逻辑错**
 ③ $left = mid$
 B. ① $a[mid] < a[mid + 1]$
 ② $right = mid + 1$ **逻辑错**
 ③ $left = mid$

- B. ① $a[mid] < a[mid + 1]$
 ② $left = mid + 1$
 ③ $right = mid$
 D. ① $a[mid] > a[mid + 1]$
 ② $left = mid + 1$ **逻辑错**
 ③ $right = mid$



- A. ① $a[mid] > a[mid + 1]$
 ② $right = mid + 1$
 ③ $left = mid$
 C. ① $a[mid] < a[mid + 1]$
 ② $right = mid + 1$
 ③ $left = mid$
 B. ① $a[mid] < a[mid + 1]$
 ② $left = mid + 1$
 ③ $right = mid$
 D. ① $a[mid] > a[mid + 1]$
 ② $left = mid + 1$
 ③ $right = mid$

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. 某空气质量监测系统负责实时采集城市各监测点的 CO₂、PM_{2.5} 数据。以 PM_{2.5} 为例，各监测点的智能终端每小时从传感器采集 60 次数据，在去除 1 个最大值和 1 个最小值后，计算剩余数据的平均值，并通过无线网络模块将数据上传至 Web 服务器。Web 服务器根据数据分析结果，经由智能终端控制声光报警器发出相应的预警信号。管理员可通过浏览器查询实时及历史数据。请回答下列问题：

(1) 响应浏览器查询请求并从数据库读取数据的程序存储在 ▲ B（单选，填字母）。

A. 智能终端 B. 服务器 C. 浏览器

(2) 该系统中，不同硬件承担的数据处理任务描述恰当的是 ▲ C（单选，填字母）。

A. 服务器接收数据并计算均值；浏览器仅呈现数据

B. 智能终端仅上传数据；浏览器计算均值并呈现数据

C. 智能终端获取传感器数据、计算均值并上传；服务器接收数据分析并存储

(3) 系统运行一段时间后，某监测点的无线网络模块损坏，可能导致的问题有 ▲ AC（多选，填字母）

A. 管理员无法查看该监测点的实时 PM_{2.5} 数据

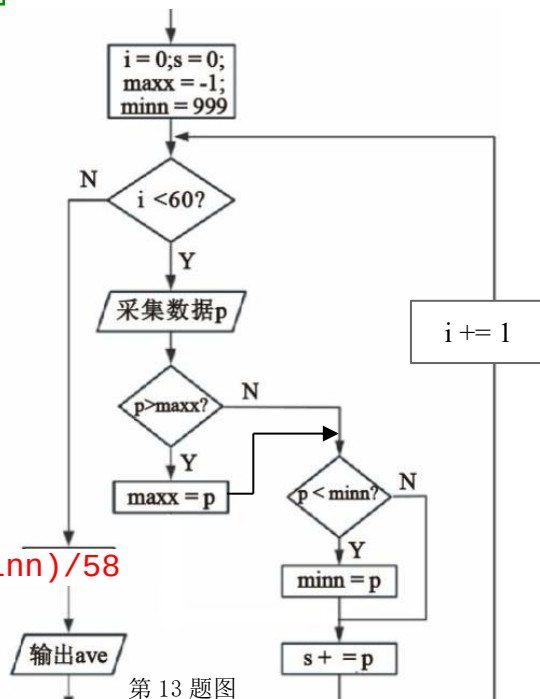
B. 服务器无法读取该监测点的历史 PM_{2.5} 数据

C. 该监测点声光报警器无法触发警报

D. 该监测点传感器无法采集 PM_{2.5} 实时数据

(4) 某个小时内智能终端数据处理的流程图如 13 题图所示，图中①处应填入 ave=(s-maxx-minn)/58

(5) 当二氧化碳浓度、PM_{2.5} 超过范围后，声光报警器将响应 web 服务器发出的预警指令，现从声光预警器角度来阐述一种方案，让使用者能分辨预警信号来自哪种数据。



第 13 题图

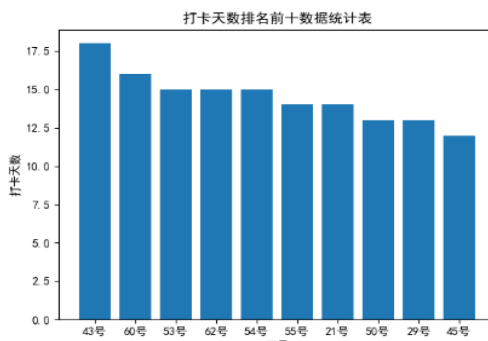
通过编码器设置报警器的工作模式，需要区别 3 种状态 例如只亮光不发出声音判断为 CO₂、只发出声音不亮光 为 PM_{2.5}，既发光又发声表示两种数据 或使用多色 LED 灯珠控制灯的颜色来区别 或设置报警器的音调来进行区别 讨论：在服务器端设置...? 设置不同报警模式（过于简单）

14. 某公司利用打卡系统进行上班打卡次数统计，9 月打卡数据导出保存在“data.xlsx”，如 14 题

图 a 所示。请回答下列问题：

工号	考勤日期	打卡时间	打卡结果
3号	25-09-01 星期一	2025-09-01 06:03	正常
3号	25-09-01 星期一	2025-09-01 08:13	异常
29号	25-09-01 星期一	2025-09-01 17:06	正常
29号	25-09-01 星期一	2025-09-01 20:36	异常
51号	25-09-29 星期一	2025-09-29 18:49	正常
51号	25-09-29 星期一	2025-09-29 21:30	正常

14 题图 a



14 题图 b

(1) 某条数据打卡时间显示为“二〇二五年九月二十六日 22:53:00”，这种情况属于 ▲ A（单选填字母：A. 格式不一致 B. 逻辑错误 C. 数据异常）

(2) 若员工每天在规定时间内仅签入一次且仅签出一次，则对应的签入签出打卡结果会显示“正

常”，否则显示异常。现统计9月份正常打卡对应的打卡天数，并将该月打卡天数排名前10的员工数据，绘制如图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）。

工号	考勤日期	打卡时间	打卡结果
3号	25-09-01 星期一	2025-09-01 06:03	正常
3号	25-09-01 星期一	2025-09-01 08:13	异常
29号	25-09-01 星期一	2025-09-01 17:06	正常
29号	25-09-01 星期一	2025-09-01 10:36	异常

```
#导入模块，设置中文显示，代码略
df = pd.read_excel("data.xlsx")
df = A ① df[df["打卡结果"] == "正常"]
df_g = df.groupby("工号",as_index = False).count()
df_sort = C ② df_g.sort_values("打卡结果",ascending = False)
df1 = df_sort.head(10)
plt.bar(df1["工号"],df1["打卡结果"]//2)
#设置绘图参数，显示如图b所示的柱形图，代码略
程序中①②处可选的代码有：
```

- A. df[df["打卡结果"] == "正常"]
- B. df["打卡结果" == "正常"]
- C. df_g.sort_values("打卡结果",ascending = False)
- D. df_g.sort_values("打卡次数")
- E. df_g.sort_values("打卡次数",ascending = False)

(3) 按原规则执行一段时间后，发现部分员工某天打卡只有签入或只有签出，因此公司设定了新的规则：打卡当日既要有签入又要有签出，且签入和签出时间间隔需满足两小时及以上才计入打卡次数。实现新规则的部分Python程序如下，请在划线处填入合适的代码。

#利用以上df对象中的数据，按照“工号”为主要关键字升序，“考勤日期”为次要关键字升序排序处理后保存至df对象中(默认每人每天只有一次签入打卡和一次签出打卡)，代码略

```
def minute(t): #提取打卡时间转成分钟
```

```
h = t[11:13]
m = t[14:16] 或 t[14:]
return int(h) * 60 + int(m)
```

工号	考勤日期	打卡时间	打卡结果
3号	25-09-01 星期一	2025-09-01 06:03	正常
3号	25-09-01 星期一	2025-09-01 08:13	异常
29号	25-09-01 星期一	2025-09-01 17:06	正常
29号	25-09-01 星期一	2025-09-01 10:36	异常

```
dk = {}
i = 0;cnt = 0
while i < len(df)-1:
    if df.at[i,"工号"] == df.at[i+1,"工号"]:
        if df.at[i,"考勤日期"] == df.at[i+1,"考勤日期"]:
            if abs(minute(df.at[i+1,"打卡时间"])-minute(df.at[i,"打卡时间"]))>= 120 ② (abs可不要)
                cnt += 1
        else:
            dk[df.at[i,"工号"]] = cnt
            cnt=0 ③
```

```
i += 1
#输出 dk 中打卡情况，代码略
```


15. 某酒店有 3 个单元楼（1 单元、2 单元、3 单元），每个单元楼有 10 层（1-10 层），配送机器人从地面（算作第 0 层）出发需按订单提交顺序选取待配送外卖，配送规则如下：
- ① 优先配送同一单元楼的外卖，凑满 3 件后配送（机器人一次性最多配送三件）；
 - ② 若同一单元楼订单不足 3 件，且后续 5 分钟内无该单元楼新订单，则直接配送当前订单；
 - ③ 配送时间计算：每层楼电梯运行时间按 1 分钟一层楼计算；抵达楼层后每配送 1 件外卖需额外 3 分钟（如 1 单元 3 层配送 1 件需 6 分钟，如 1 单元 3 层、5 层各 1 件，共需 11 分钟，不同楼层须按楼层由低到高依次送达）。请回答下列问题：
- （1）已知某时段外卖订单按提交顺序排列如下表，提交时间已转换为分钟，例 840 分钟表示 14 点整。初始无待配送订单。根据配送规则，第 1 次 1 单元配送的订单需 11 分钟。

订单编号	提交时间（分钟）	单元楼	楼层
D01	840	<u>1单元</u>	3层
D02	841	2单元	5层
D03	842	<u>1单元</u>	5层
D04	843	3单元	8层
D05	844	<u>1单元</u>	2层
D06	846	2单元	7层
D07	847	1单元	6层

链队

每条链标记
链头、链尾

```
while head < len(data) :
```

if que[k-1][1]==-1: 当前链 为空链, 成为链头

else:

```
head+=1
```

def proc(qs, ucnt): #列表 ucnt 存储各单元待配送订单数量

```
for i in range(1,n): 冒泡 双关键字排序 排序结果：排在首的单元的订单先
    for j in range(n-i): 处理
```

$$\overline{\text{ucnt}[\text{j}], \text{ucnt}[\text{j}+1]} = \text{ucnt}[\text{j}+1], \text{ucnt}[\text{j}]$$

```

return unit

```

A. $\text{ucnt}[j] > \text{ucnt}[j+1]$ or $\text{ucnt}[j] == \text{ucnt}[j+1]$ and $\text{data}[\text{qs}[\text{unit}[j]-1][0]][1] > \text{data}[\text{qs}[\text{unit}[j+1]-1][0]][1]$

C. ucnt[j]<ucnt[j+1] or ucnt[j]==ucnt[j+1] and data[qs[unit[j]-1][0]][1]>
data[qs[unit[i+1]-1][0]][1] 数量相同按 每个单元楼首单时间升序排列

(3) 实现机器人配送功能 Python 程序如下。请在划线处填入合适的代码。

```
def delivery(d, curtime):    #指定单元楼单次配送
    waittime = 5; total = 0 #统计配送耗时
    p = que[d-1][0]
    temp = [] #存储本次配送的订单索引 机器人本次的任务列表
    ft = data[p][1] 本次机器人开始时间 first time
    while p != -1:
        temp.append(p)
        if data[p][4] != -1: p订单之后的订单时间
            st = data[data[p][4]][1]
        else:
            st = -1
        wt = st - ft
        if len(temp) >= 3 or wt >= waittime or p == que[d-1][1]:
            maxf = 0
            for i in temp:
                if data[i][3] > maxf: 找到最高楼层, 算出电梯时间
                    maxf = data[i][3]
            que[d-1][0] = data[temp[-1]][4] 或 que[d-1][0] = data[p][4]
            或 que[d-1][0] = data[temp[len(temp)-1]][4]
            total = maxf + 3 * len(temp)
            curtime += total 刷新curtime, 也就是机器人下一次可以派送的时间
            break → 代码修正, 添加: ucnt[0] -= len(temp) 刷新本单元订单数
        p = data[p][4]
    return total, curtime
curtime = data[0][1]; queIn(head)
#将各单元待配送订单数量存入 ucnt, 代码略
while not (que[0][0] == -1 and que[1][0] == -1 and que[2][0] == -1):
    unit = proc(que, ucnt)
    d = unit[0] 排序后, unit[0] 就是机器人首先要处理的单元楼
    total, curtime = delivery(d, curtime)
    print("本次配送的单元是: ", d, "共耗时" + str(total) + "分钟")
```

派送规则:

1. 凑3个订单
2. 本次首单后续5分钟无订单则马上派送
3. 若尾单也马上派送(隐藏要求)