

技术（三）

本试卷分两部分，第一部分信息技术，第二部分通用技术。满分 100 分，考试时间 90 分钟。

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、多选、错选均不得分）

阅读下列材料，回答第1至6题：

智能CT影像系统是基于计算机断层扫描（CT）技术，结合人工智能、大数据和网络通信的医疗诊断系统。该系统能够高效采集、存储和分析患者的CT影像数据，并利用AI算法辅助医生进行病灶检测（如肿瘤、炎症等），提高诊断准确性和效率。

- 关于智能CT影像系统中数据的说法，正确的是
 - CT影像数据能用于诊断疾病，体现了数据的价值性
 - CT影像数据分析处理后不再以二进制形式存储
 - CT影像数据是结构化数据
 - CT影像数据仅存在于医院服务器中
- 关于信息安全与信息社会责任，下列行为合适的是
 - 同一科室医生共用一个系统账户
 - 未经同意即公布病人的检测情况
 - 定期备份检测数据
 - 随意剪辑病人CT影像
- 下列关于该信息系统组成的说法，不正确的是
 - CT扫描仪属于输入设备
 - 病人和医生都属于信息系统用户
 - 智能CT影像系统软件是应用软件
 - 服务器的存储器容量不会影响系统性能
- 该系统中人工智能是基于神经网络方法实现的，下列说法正确的是
 - 当AI的诊断结果和医生的诊断结果冲突时，应以AI的诊断结果为准
 - 该人工智能需要预先手工设置所有疾病的知识库
 - 提升服务器性能、优化AI算法、提升CT拍摄质量都有助于提高人工智能诊断准确率
 - 该人工智能进行病症诊断时，需要原始的训练数据进行比对
- 下列技术中，最可能用于连接 CT 扫描仪和医院服务器的是
 - 3G
 - 网线
 - 蓝牙
 - RFID

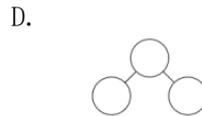
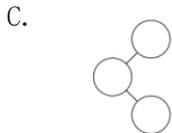
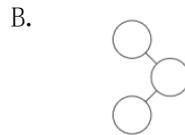
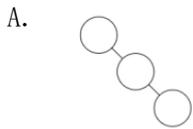
6. 将CT影像数据存储为未经压缩的BMP格式文件，下列说法不正确的是

- A. 影像数据采集需经过采样、量化和编码过程
- B. 病人病情越复杂，得到的单张图像文件的存储容量越大
- C. 将BMP格式转换成JPG格式有助于节省存储空间
- D. CT影像共256色，则BMP文件量化位数为8位

7. 某栈初始有 3 个元素，经过一系列入栈、出栈操作后，栈变为空，若元素出栈顺序为 1, 2, 3, 2，则栈初始从栈底到栈顶的元素值不可能是

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

8. 某二叉树前序遍历结果为 ABC，后序遍历结果为 CBA，其树形结构图不可能是



9. 有如下 python 程序段：

```
dic={}
s="a12b4c563"
t=0
for i in s:
    if "0"<=i<="9":
        t=t*10+int(i)
    else:
        dic[i]=t
        t=0
```

则程序运行后，dic["b"]的值为

- A. 2
- B. 12
- C. 4
- D. 563

10. 下列 python 程序段用于输出 a, b, c 三个变量中第 2 大的值

```
if a>b:
    if b>c:
        print(b)
    elif a>c:
        print(c)
    else:
        print(a)
else:
    if c>b:
        print(b)
    elif a>c:
        print(c)
    else:
        print(a)
```

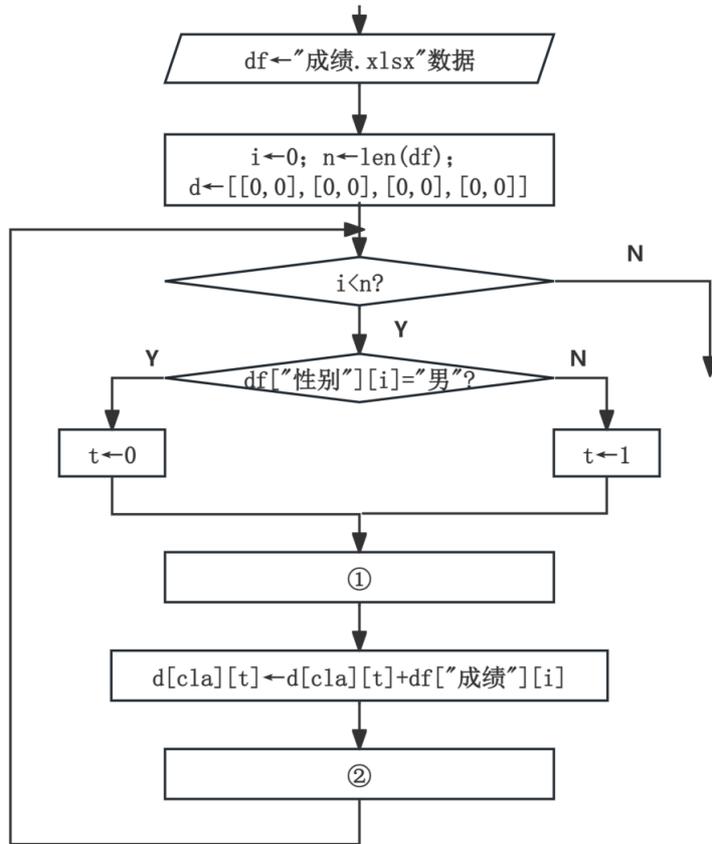
程序存在漏洞，下列数据能测试出问题的是

- A. a=1 ; b=2 ; c=3
 - B. a=1 ; b=2 ; c=1
 - C. a=1 ; b=3 ; c=2
 - D. a=3 ; b=1 ; c=2
11. 有如下 python 程序段:

```
def sorts(a, st, ed):
    if st==ed:
        return
    for i in range(ed, st, -1):
        if a[i]>a[i-1]:
            a[i], a[i-1]=a[i-1], a[i]
    sorts(a, st+1, ed)
sorts(a, 0, 4)
```

若列表 a 初值为[4, 3, 1, 7, 6]，则运行程序后，a 的值变为

- A. [7, 6, 4, 3, 1]
- B. [1, 3, 4, 6, 7]
- C. [1, 3, 4, 7, 6]
- D. [4, 7, 6, 3, 1]



第 13 题图 b

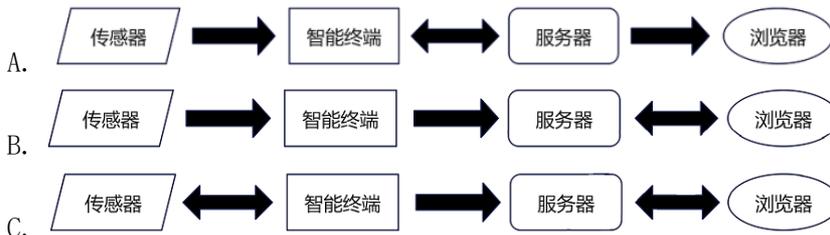
- (1) 请在第 13 题图 b 中①②处填入合适内容。
 (2) 下列代码也能实现本功能的是 ▲ (单选, 填字母)。

- A. `df1=df.groupby("班级")`
`df2=df1.groupby("性别").sum()`
 B. `df["组"]=df["班级"].astype(str)+df["性别"]`
`df1=df.groupby("组").sum()`

#注: `astype(str)`将数字列转换为字符串列

14. 某高中搭建教室温度监控系统, 该系统能监测该校 n 个教室 (分别编号 $0 \sim n-1$) 的温度。每个教室装有温度传感器和智能终端, 传感器采集的数据由智能终端经 IoT 模块发送到 Web 服务器, 用户通过浏览器可快速查询某编号教室的平均温度。请回答下列问题:

- (1) 本系统中传感器、智能终端、服务器与浏览器之间数据的传输关系, 其中合理的是 ▲ (单选, 填字母)。



(2) 某个智能终端的部分代码如下:

```
class=102
while True:
    temp=pin0.read_analog()
    Obloq.get("con?id="+str(class)+"&temp="+str(temp),10000)
    sleep(1000)
```

若获取温度为 30, 则其提交数据到 Web 服务器的 URL 为 `http://192.10.152.197/▲`。

(3) 服务器上的程序具有功能 a: 获取并存储各个智能终端传递来的教室编号和温度; 功能 b: 获取浏览器传递来的教室编号后快速查询该教室的平均温度。请在程序中划线处填入合适的代码。

功能 a 部分代码如下:

```
head=[-1]*n ; data=[] ; i=0
```

```
while True:
```

```
    #获取智能终端传递的编号和温度, 保存在 id 和 temp 中, 代码略
    data.append([id, temp, head[id]])
```

①

```
    i+=1
```

功能 b 部分代码如下:

```
#获取浏览器传递来的查询编号, 保存在 id_find 中, 代码略
```

```
p=head[id_find]
```

```
m=s=ans=0
```

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

②

```
    m+=1
```

```
    p=data[p][2]
```

```
if m!=0:
```

```
    ans=s/m
```

(4) 运行一段时间后, 管理员发现服务器仅无法获取编号 4 教室的温度数据, 系统中造成上述问题的原因可能是 ▲ (经检测, 各器件之间的连接均正常)

15. 扫雷游戏规则如下：一个 $n \times n$ 大小的棋盘，共有 m 个方格上为雷。游戏开始时，所有方格为未揭示状态，选择方格来揭示它的内容，若为雷则游戏失败；若为数字，则表示周围一圈 8 个方格中（边界上方格的周围格子数会小于 8）雷数量，例如第 15 题图 a 所示。如果被揭示的方格为数字 0，那么该方格会展开，继续揭示周围一圈方格。玩家揭示雷则游戏失败，揭示所有非雷方格才游戏胜利。

(1) 某棋盘如第 15 题图 b 所示，已有 4 个格子被揭开，可推测 A 点内容为 ▲（单选，填字母：A. 雷 / B. 0 / C. 1 / D. 2）。

雷	雷	1
2	3	2
0	1	雷

第 15 题图 a

1	2	
	3	
1		A

第 15 题图 b

(2) 编写扫雷游戏程序，定义 `outner` 数组用于存储游戏的显示界面，仅包含“#”和数字，“#”表示方格未揭示，数字表示已揭示方格的周围雷点数。打印显示界面并判断游戏是否胜利的 `pri` 函数如下：

```

n=m=10
outner=[["#" for i in range(n)]for j in range(n)]
def pri():
    cnt1=cnt2=0
    for i in range(n):
        for j in range(n):
            print(outner[i][j],end=" ")
            if "0"<=outner[i][j]<="8":
                cnt1+=1
            if outner[i][j]=="#":
                cnt2+=1
        print()
    if :
        print("恭喜，你胜利了")
        return True
    else:
        return False

```

加框处代码用于判断游戏是否胜利，下列语句正确的是 ▲（多选，填字母）。

- A. $cnt1+m==n*n$
- B. $cnt1+cnt2==n*n$
- C. $cnt2==m$
- D. $cnt2<m$

(3) 实现上述功能的部分 Python 程序如下，其中 `inner` 数组用于存储各个方格的周围雷点数，若值为 -1 表示为雷，请在划线处填入合适的代码。

```

inner=[[0 for i in range(n)]for j in range(n)]
def generate():      #生成初始棋盘
    #随机生成 m 个雷点位置, 统计并生成 inner 数组的值, 代码略
def judge(x,y):    #根据选中的位置, 判断结果
    if inner[x][y]==-1:
        print("抱歉, 你失败了")
        return False
    elif _____①_____:
        outner[x][y]=str(inner[x][y])
    else:
        que=[[-1,-1]for i in range(n*n)]
        que[0]=[x,y]
        head=0
        tail=1
        while head<tail:
            cur=que[head]
            outner[cur[0]][cur[1]]=str(inner[cur[0]][cur[1]])
            w=[[-1,-1],[-1,0],[-1,1],[0,-1],[0,1],[1,-1],[1,0],[1,1]]
            for step in w:
                _____②_____
                if 0<=next[0]<n and 0<=next[1]<n and next not in que:
                    x=inner[next[0]][next[1]]
                    if x==0:
                        que[tail]=next
                        tail+=1
                    outner[next[0]][next[1]]=str(x)
                _____③_____
        return True
generate()
while not pri():
    x=int(input("请输入点击节点的横坐标(0~n-1):"))
    y=int(input("请输入点击节点的纵坐标(0~n-1):"))
    if not judge(x,y):
        break

```