

高三上信息选考练习

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 4 题：

某校基于线下文印室衍生出在线数字文印通。用户通过终端浏览器登录，上传文档、表格、图片等数据，平台将数据分类存储后安排相关文印任务。文印通为用户提供费用结算、任务查询、历史记录检索、优惠活动推送等功能。用户还可向智慧助手以语音、图片、文字的形式咨询相关问题。

1. 关于该数字文印通中数据的叙述，正确的是 **D**
- A. 该文印通中的所有数据都是结构化数据
 B. 用户上传的文本、表格、图片等都是信息
 C. 不同类型的数据可以脱离载体直接上传至文印通
 D. 分类存储可以有效地管理数据，提高了文印效率
2. 下列有关信息安全与保护的做法，合理的是 **C**
- A. 文印通升级时，必须关闭防火墙
 B. 以明文方式公开用户每次的支付账单
 C. 需要用户通过手机号和口令验证身份登录
 D. 未经允许将用户已打印过的文件用于商业交易
3. 下列对文印通中数据处理的说法，不合理的是 **A**

文本、表格、图片等是数据

- A. 为了提高图片的打印质量，降低图片的总像素数
 B. 文印费用结算体现了数据加工处理的功能
 C. 为了便于分享，将登录该文印通的网址制作成二维码
 D. 该文印通中所有数据以二进制方式存储在计算机中
4. 该文印通的下列应用中，体现人工智能技术的是 **D**
- A. 将用户上传的数据保存到数据库中
 B. 提供打印任务状态和历史记录查询功能
 C. 在主页不定期推送相关的文印优惠活动
 D. 通过智慧助手回答用户提交的文印相关问题

图片总像素数决定了图片的清晰度，降低像素数会导致打印质量下降，而非提高

阅读下列材料，回答第 5 至 6 题：

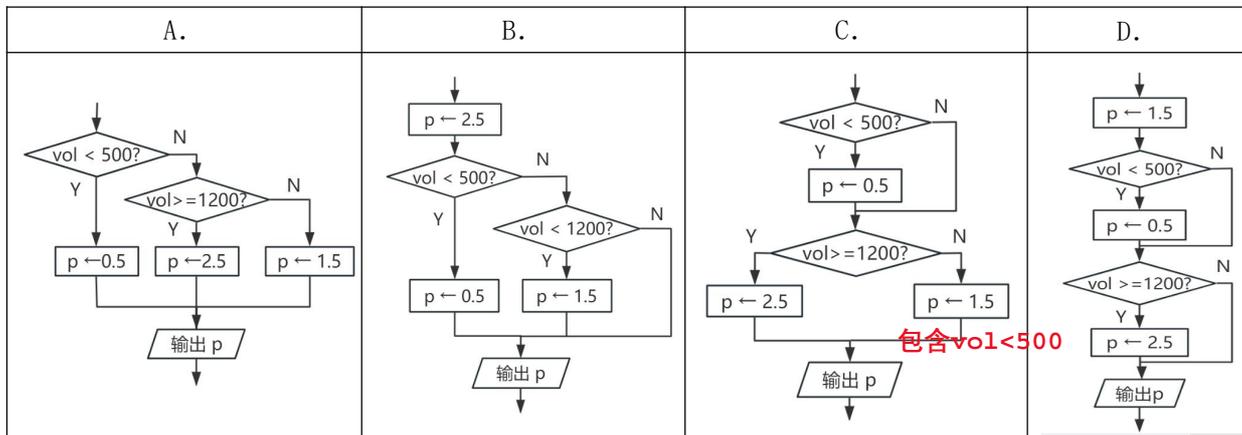
某小区采用新型健身房管理系统，业主可通过刷门禁卡或人脸识别的方式进入场馆。各运动器材内置传感器收集运动者的体能数据，传输至服务器分析数据，生成运动报告。用户可随时随地登录该系统 app 查询实时人流量及阅读运动报告。

5. 关于该系统设计与应用的说法，正确的是 **A**
- A. 利用人流量数据分析健身的高峰时段
 B. 进入场馆仅运用了 RFID 技术
 C. 系统中的硬件包括门禁卡、器材、app
 D. 系统运行需要传感技术的支持，不需要控制技术
6. 下列关于网络技术与通信的说法，合理的是 **C**
- A. 该系统中可以共享的网络资源不包括软件资源
 B. 使用手机登录系统，必须使用移动通信网络
 C. 服务器网络故障将影响运动质量报告的生成
 D. 运动器材通过 WiFi 接入网络后，无需 IP 地址也能与服务器正常通信

还有人脸识别（生物识别技术）

计算机网络也可以

7. 某许愿星瓶子重量 500 克(不包含)以下 0.5 元/个, 500 克到 1200 克(不包含)1.5 元/个, 1200 克及以上 2.5 元/个。若重量为 vol (单位: 克), 价格为 p (单位: 元), 执行下列部分流程图后, 输出 p 的值不符合要求的是 **C**

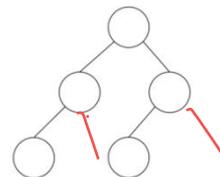


8. 利用栈求逆波兰表达式的方法: 从左往右扫描该表达式, 遇到数字时入栈; 遇到运算符时, 把处于栈上方的两个元素依次出栈, 用运算符计算, 并把结果压入栈中。如此反复操作, 直至表达式扫描结束。使用该算法求表达式 “3 7 2 - + 4 * 8 /” 的值时, 所使用的栈容量至少为 **B**

- A. 2 B. 3 C. 4 D. 5

9. 某二叉树的树形结构, 如第 9 题图所示, 下列说法正确的是 **D**

- A. 该二叉树是一棵完全二叉树
 B. 该二叉树的深度为 3, 度为 2 的节点有 2 个
 C. 若该二叉树中序遍历为 DBAEC, 则前序遍历为 ABCDE
 D. 若将其补全为高度是 4 的满二叉树需再添加 10 个节点



第 9 题图

10. 有如下 Python 程序段:

```
def dig(n, k):
    k = k - 1                      #①
    if k == 0:
        return n % 10
    else:
        return dig(n // 10, k)    #②
```

求倒数第 k 位数

```
print(dig(52617, 3))
```

$$dig(52617, 3) = dig(5261, 2) = dig(526, 1) = 6$$

下列说法不正确的是 **D**

- A. 执行该程序段后, 输出结果为 6
 B. 执行该程序段, 函数 dig 一共被调用了 3 次
 C. $dig(n, k)$ 函数的功能是输出 n 中从右往左数第 k 个元素
 D. 若删除语句①, 将语句②改成 $return dig(n//10, k-1)$, 不影响程序运行结果

程序功能是生成 5 个 [1,10] 的随机整数，按从大到小的顺序输出（统计每个数字出现次数，反向遍历统计数组并输出）。

11. 有如下 Python 程序段：

```
import random
n = 5; a = [0] * (n + 1); c = [0] * 11
for i in range(1, n + 1):
    a[i] = random.randint(1,10)           #随机生成[1,10]范围内的整数
    c[a[i]] += 1
for i in range(len(c)-1, 0, -1):
    for j in range(c[i]):
        print(i, end = " ")
```

桶排

执行该程序段后，输出的结果可能为 **B**

- A. 1 1 2 3 7 B. 8 6 4 3 2 C. 5 7 9 6 8 D. 10 8 6 8 2

12. 某 Python 程序段如下：

```
s = "21027,53523,042,"
q = [0] * 100; head = tail = 0; tmp = ""; flag = True
for i in range(len(s)):
    if "0" <= s[i] <= "9":
        q[tail] = s[i]; tail += 1
    elif s[i] == ",":
        while head < tail:
            tmp += q[head]; head += 1
            if head != tail and flag :
                head += 1
        flag = not flag
```

T F
21027 53523 042
✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓

跳过一位

执行该程序段后，变量 tmp 的值为 **B**

- A. "20735204" B. "20255202" C. "20755302" D. "20235304"

二、非选择题（本大题共 2 小题，其中第 13 小题 10 分，第 14 小题 9 分，共 19 分）

13. 某研究小组搭建了一个校园车辆速度监测系统。该系统设有若干个监测点，智能终端通过 IoT 模块将采集的车速数据上传至 Web 服务器，若车速超过阈值达到 3 次及以上，则该车牌会被系统拉黑，将不会被识别，车辆无法进入校园。管理员可通过浏览器查看数据。请回答下列问题：

(1) 该小组要完成该系统的搭建，需要编写的程序是 **A**▲（单选，填字母：A. 服务器端程序/ B. 客户端程序）

(2) 下列关于该系统智能终端的说法，正确的有 **BD**▲（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. 通过浏览器查看历史数据需访问智能终端 **访问 服务器**
- B. 传感器与智能终端之间的数据传输是单向的 **传感器-智能终端**
- C. 若服务器故障，智能终端无法继续采集传感器数据
- D. 智能终端具备数据存储和处理能力 **例如 电子秤 根据重量和金额直接计算总价**

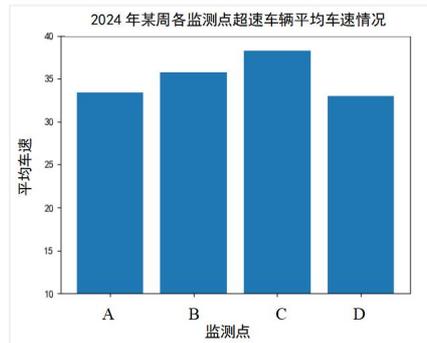
(3) 若提交数据到 Web 服务器的 URL 为 `http://192.168.10.2:8080/input?id=3&speed=27`, 则服务器端与该 URL 关联的路由设置语句是 `@app.route("/input")`

(4) 为了让用户遵循限速规定, 避免超速行为, 顺利进出校园, 请选择其中一个角度提出一条合理的功能设计建议。
 通过短信的方式发给用户超速次数信息; 或为管理员增设取消拉黑操作的权限, 恢复车辆通行;

(5) 该小组将某周的数据导出, 部分数据如第 13 题图 a 所示。车辆行驶速度超过 30 (单位: 千米/小时) 则判定为超速, 现要分析每个监测点超速车辆行驶的平均车速情况, 并绘制如第 13 题图 b 所示的柱形图, 部分 Python 程序如下:

#	A	B	C	D
1	车牌号码	监测点	车速	时间
2	浙J10000	A	5	2024-12-02 06:31:26
3	浙J20000	C	14	2024-12-02 07:24:15
4	浙J30000	B	9	2024-12-03 07:44:26
5	浙J40000	C	33	2024-12-04 08:19:09
707	浙J50000	A	8	2024-12-06 15:33:01
708	浙J60000	D	7	2024-12-07 19:08:16
709	浙J70000	B	10	2024-12-07 20:15:21

第 13 题图 a



第 13 题图 b

```
import pandas as pd
import matplotlib.pyplot as plt
# 显示中文字符, 设置字体, 代码略
df = pd.read_excel("car.xlsx")
df1 = df[df.车速>30]
df2 = df1.groupby("监测点").mean()
plt.bar(df2.index, df2.车速)
# 设置绘图参数, 显示如第 13 题图 b 所示的柱形图, 代码略
print(df4)
```

监测点	车速
A	40.0
B	39.6
C	44.5

①请在程序中划线处填入合适的代码。
 ②若要查看这一周被系统因超速拉黑的次数最多的前 10 条车辆数据记录, 方框中应填入的语句依次为 ABD (选 3 项, 填字母序列, 少选、多选、错选或次序错均不得分)。

- A. `df3 = df1.groupby("车牌号码", as_index=False).车速.count()`
- B. `df3 = df3[df3.车速>=3]`
- C. `df3 = df3[df3.车牌号码>=3]` **groupby后, 车牌号码是索引! 车速列才是统计数据**
- D. `df4 = df3.sort_values("车速", ascending=False).head(10)` **降序取前10**
- E. `df4 = df3.sort_values("车速")[:10]`

14. 某考试在某个考点共有 n 名考生, 每个试场共 m 个座位, 每位考生的信息包含准考证号、姓名。按照所有考生的准考证号由小到大依次编排试场号和座位号 (为增删方便, 数据以链表的形式存储)。编写程序模拟查找过程, 输入考生准考证号, 输出考生信息。

若 n 为 60, m 为 20, 编排后的考生信息如第 14 题图 a 所示, 以准考证号升序构成链接关系后, 进行分段索引查找, 每段长度为 m , 每段结束位置的下标存入列表 b , 列表 b 的结果为 $[23, 58, 42]$ 。若最后一个试场人数不足 m , 则不用记录结束位置。程序运行结果如第 14 题图 b 所示。请回答下列问题:

#	A	B	C	D
1	准考证号	姓名	试场号	座位号
2	2501178	李悦	113002	12
3	2501156	王梓轩	113002	7
24	2501074	谷杨雨	113001	13
25	2501114	刘梓涵	113001	20
44	2501322	赵俊成	113003	20
45	2501112	陶瑞	113001	19
59	2501221	牟志霆	113003	1
60	2501213	徐一展	113002	20
61	2501307	施曼曼	113003	16

第 14 题图 a

请输入准考证号: 2501156
姓名: 王梓轩 试场号: 113002 座位号: 7

第 14 题图 b

(1) 当 n 为 80, m 为 30 时, 则列表 b 中的元素个数为 2 ▲ (填数字)

(2) 定义如下 linksort(data, mark) 函数, 参数 data 和 mark 共同模拟链表的结构, data 存储每位考生的准考证号, mark 表示 data 中每个元素指向下一个元素位置的指针。函数功能是使链表按照准考证号升序排列, 最后返回头指针 head。

① 若 data 为 [2501067, 2501078, 2501046, 2501059, 2501044], mark 为 [-1, -1, -1, -1, -1], 调用 linksort(data, mark) 函数后, 返回 head 的值为 4, mark 的值为 [1, -1, 3, 0, 2]。

② 请在划线处填入合适的代码。

data 中没有指针域, 利用 mark 实现指针域功能

```
def linksort(data, mark):
```

链表排序经典方法: 插入排序

```
    head = 0
```

```
    for i in range(1, len(data)):
```

```
        p = head
```

```
        if data[p] >= data[i]:
```

```
            mark[i] = p
```

```
            head = i 找最小值作为链头
```

```
        else:
```

```
            while p != -1 and data[i] > data[p]: 将 i 节点插入到 p 节点之后
```

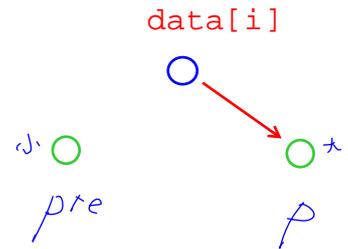
```
                pre = p
```

```
                p = mark[p]
```

```
            mark[i] = p
```

```
            mark[pre] = i
```

```
    return head linksort 是函数, 返回 head!
```



(3) 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。

```
def search(key, data, mark):
```

```
    i = 0; j = k - 1 b = [-1] * k, 由此可知 i, j 是针对 b 数组的左右边界
```

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

```
        mid = (i+j)//2
```

```
        if key <= data[b[mid]]:
```

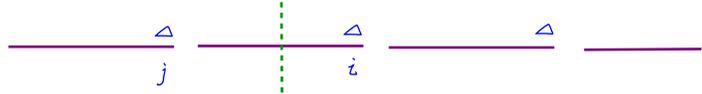
```
            j = mid - 1
```

```
        else:
```

```
            i = mid + 1
```



若 n 为 60, m 为 20, 编排后的考生信息如第 14 题图 a 所示, 以准考证号升序构成链接关系后, 进行分段索引查找, 每段长度为 m, 每段结束位置的下标存入列表 b, 列表 b 的结果为 [23, 58, 42]。若最后一个考场人数不足 m, 则不用记录结束位置。程序运行结果如第 14 题图 b 所示。请回答下列



```

if i == 0:
    p = head
    q = b[i]
elif i == k:
    p = mark[b[k-1]]
    q = -1
else:
    p = mark[b[i-1]] 或 p = mark[b[j]]
    q = b[i]
while p != q and key > data[p]:
    p = mark[p]
if key == data[p]:
    return p
else:
    return -1

```

在第一个考场里找, p 是该考场的头, q 是该考场的尾

在最后一个考场里找, p 是该考场的开始位置, q 是链尾

在中间的那些考场里找, 要找的是 i 考场, b[i] 是尾, 头是 b[i-1] 或 b[j] 的下一个节点, 即 mark[b[i-1]]

p、q 之间用顺序查找的方式比对

未找到, 则返回 -1

""" 读取所有考生的信息存入二维列表 message, 如 [[2501178, '李悦', 113002, 12], [2501156, '王梓轩', 113002, 7], [2501064, '周逸飞', 113001, 11], ...], 从 message 中提取准考证号存入 data 列表, 如 [2501178, 2501156, 2501064, ...], 读取考生总人数存入 n, 每个试场的座位数存入 m, 代码略 """

```

mark = [-1] * n
k = n//m; b = [-1] * k
head = linksort(data, mark)
p = head
for i in range(k):
    for j in range(m-1):
        p = mark[p]
        b[i] = p
        p = mark[p]
key = int(input("请输入准考证号: "))
p = search(key, data, mark)
# 输出考生信息、试场号和座位号, 代码略

```

记录第 i 个考场最后一个考生的下标

本题思路:

单向链表 查找数据基本方式: 顺序查找

为了加快顺序查找速度, 设置若干标志点, 将链表分为若干区段;

顺序查找限定在某个具体区段内查找