

2023年6月信息高考真题

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)

1. 下列关于数据和信息的说法,正确的是 **D**
- A. 在数据处理过程中不会有新的信息产生 **X**
 - B. 信息的保存和传播可以不依附于载体 **X**
 - C. 信息的价值因人而异,但信息不会有虚假 **X**
 - D. 计算机中的数据表现形式不同,但都以二进制方式存储

阅读下列材料,回答第 2 至 4 题:

某智慧课堂系统的部分功能是:教师进教室后刷校园一卡通实现身份认证并启动系统,学生进教室时通过摄像头刷脸签到,签到结果保存在服务器的数据库中,系统可自动生成考勤报表。课堂教学中,教师可将教学资源发送到学生的移动终端,学生可将作业文档上传到系统。可以现场录制教学视频并保存到服务器,系统每天定时备份数据。

2. 下列关于该系统功能和应用的说法,不正确的是 **A**
- A. 该系统设计合理,不存在局限性 **X**
 - B. 教学视频保存到服务器,有利于师生跨时空学习
 - C. 该系统具有数据采集、处理和存储等功能
 - D. 数据库可以用于存放该系统中的教师身份数据
3. 该系统的下列应用中,体现人工智能技术的是 **C**
- A. 将现场录制的教学视频保存到服务器
 - B. 系统自动生成考勤报表
 - C. 学生进教室时通过摄像头刷脸签到
 - D. 教师将教学资源发送到学生的移动终端
4. 下列关于信息系统安全和信息社会责任的说法,正确的是 **C**
- A. 教师刷一卡通实现身份认证,能确保系统没有安全隐患 **X**
 - B. 系统服务器若安装了杀毒软件,数据文件就不会被病毒侵害 **X**
 - C. 系统每天定时备份数据,是保护数据安全的重要措施
 - D. 未经授权,可将课堂教学视频放到电商平台出售 **X**

5. 下列关于网络系统的说法,不正确的是 **D**
- A. 局域网内部通信需要遵循网络协议
 - B. 局域网内可以同时有无线通信和有线通信两种方式
 - C. 校园网内可以搭建多个局域网
 - D. 可用作服务器的计算机无法用作客户端 **都可以用**
6. 某智能手机安装了鸿蒙操作系统,其主板集成了处理器、存储器等芯片,下列说法正确的是 **D**
- A. 存储器可以存放用户数据而不能存放应用程序
 - B. 鸿蒙操作系统是智能手机重要的应用软件 **系统软件**
 - C. 处理器性能是智能手机性能的唯一指标 **内存,分辨率**
 - D. 智能手机硬件包括输入、处理、存储和输出等部分

7. 对某段声音进行数字化,量化值的范围是 0~200,则量化位数最少是 **B**
- A. 7
 - B. 8
 - C. 16
 - D. 200

8. 某二叉树的树形结构如第 8 题图所示,其前序遍历结果为 BDEFCA,则中序遍历结果为 **A** **第8题图**
- A. EDCFBA
 - B. ECFDAB
 - C. BFDEAC
 - D. EDFCBA

9. 栈 s 的**最大长度为 3**,初始为空,经过一系列入栈、出栈操作,若元素入栈的顺序是 a, b, c, d, e, f, 则可能的出栈序列为 **B**

- 度为6** A. f, e, d, c, b, a **B** B. c, b, a, f, e, d C. c, a, b, d, e, f D. c, e, d, b, a, f

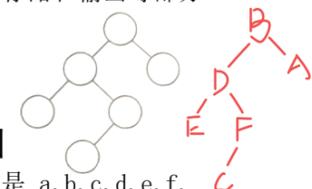
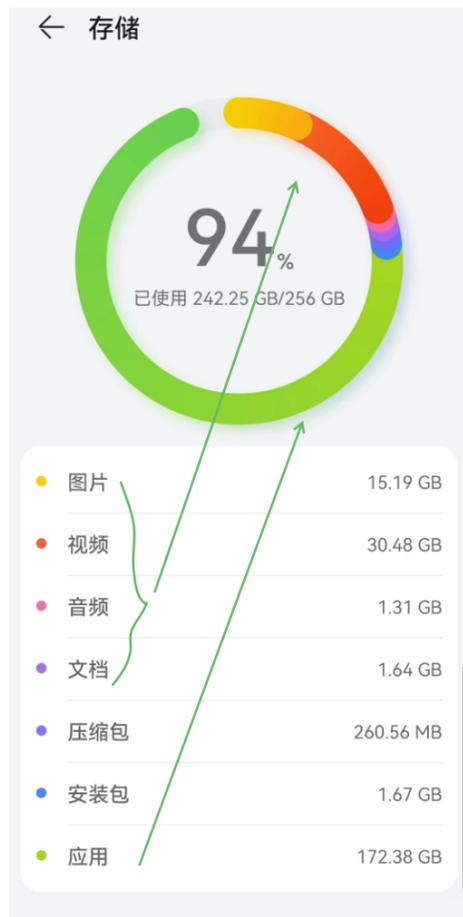
10. 定义如下函数:

```
def f(a, s):
    if a >= s:
        return a
    else:
        return f(a+1, s-a)
```

f(6,21)=f(7,15)=f(8,8)=8

- 执行语句 k=f(6, 21)后, k 的值为 **C**
- A. 6
 - B. 7
 - C. 8
 - D. 9

11. 列表 q 长度为 20, q[0]至 q[4]的值依次为'p', 'r', 'i', 'n', 't', 执行如下程序段后,输出的最后一个字符为 **D**

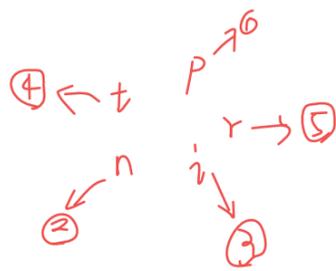


检查
1. 符合出入栈规则
2. 栈长度3

```

head,tail = 0,5
while head < tail:
    if head % 3 == 0:
        print(q[head])
    else:
        q[tail] = q[head]
        tail += 1
        head += 1

```



A.t B.n C.i D.r

12. 已排序的列表 a 有 n 个整型元素, 现要查找出现次数最多的值并输出。若出现次数最多的值有多个, 则输出最前面的一个。实现该功能的程序段如下, 方框中应填入的正确代码为 **A**

```

c,m,v=1,1,0
for i in range(1, n):
    [ ]

```

print(a[v])

<pre> A.if a[i]==a[i-1]: c += 1 if c > m: m = c v = i else: c = 1 </pre>	<pre> B.if a[i]==a[i-1]: c += 1 if c > m: m = c v = i else: c = 1 </pre>	<pre> C.if a[i]==a[i-1]: c += 1 else: if c > m: m = c v = i-1 c = 1 </pre>	<pre> D.if a[i]==a[i-1]: c += 1 else: if c > m: m = c v = i-1 c = 1 </pre>
---	---	---	---

二、非选择题(本大题共 3 小题, 其中第 13 小题 7 分, 第 14 小题 10 分, 第 15 小题 9 分, 共 26 分)

13. 某仓库有一排连续相邻的货位, 编号依次为 0~n-1, 用于放置 A, B 两种类型的箱子, A 型箱子占 2 个相邻货位, B 型箱子占 1 个货位。编写程序, 根据已完成的放置或搬离操作, 输出空货位数及还可放置 A 型箱子的最多数目(不移动已放置的箱子)。请回答下列问题:

箱子类型	操作类型	货位编号
B	放置	5
A	放置	2, 3
B	放置	0
A	放置	7, 8
A	搬离	2, 3

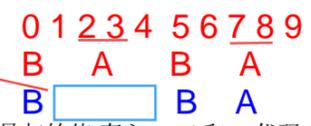
(1) 若 n 为 10, 开始时货位全空, 经过如第 13 题图所示的放置或搬离操作后, 不移动已放置箱子的情况下, 还可放置 A 型箱子的最多数目为 **2** 个

(2) 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。

```

# 读取货位总数, 存入 n, 代码略
cnt1 = n
lst = [0]*n # 货位状态, 0 表示对应的货位为空
while True:
    # 读取本次已操作的数据: 箱子类型、操作类型、货位编号起始值, 存入 t、d 和 s, 代码略
    if t == 'A':
        w = 2
    elif t == 'B':
        w = 1
    else:
        break
    if d == 'P':
        cnt1 -= w
    else:
        cnt1 += w
    lst[s] = 1 - lst[s]
    if t == 'A':
        lst[s+1] = 1 - lst[s+1]
    i, cnt2 = 0, 0
    while i < n-1:
        if lst[i] == 0 and lst[i+1] == 0:
            i += 1
            cnt2 += 1
        i += 1
    print('当前空货位数:', cnt1, ', 还可放置 A 型箱子的最多数目:', cnt2)

```



A 占 2 格, B 占 1 格

空位变多, 所以 d == 'P' 表示放入箱子, 空位减少
不管是放入还是拿走, 操作后该位置的值反向
若是 A 箱子, 后一个位置的值也反向

存在连续两个空位, A 的放置空间 cnt2+1, 循环变量 i 需要走 2 步

14.小华要搭建书房环境监控系统。该系统能实时监测书房温度和湿度，出现异常时发出警报；用户可通过浏览器查看实时监测结果和历史数据。小华已选择的硬件有：智能终端、温湿度传感器、执行器(如蜂鸣器)、服务器等，系统的硬件搭建方式是：服务器通过无线网络连接智能终端，智能终端连接传感器和执行器。请回答下列问题：

(1)该系统中,智能终端与服务器端之间的数据传输 C (单选, 字母: A.只能由智能终端到服务器端 / B.只能由服务器端到智能终端 / C.既可以由智能终端到服务器端也可以由服务器端到智能终端)。

(2)下列功能需要在智能终端程序中实现的是 A (单选, 填字母: A.采集温湿度传感器上的数据 / B.处理浏览器访问请求)。**→服务器的工作**

(3)小华基于 Flask Web 框架编写服务器端的程序，部分代码如下。编写完成后，若要通过浏览器获取视图函数 index() 的返回页面，则应访问的 URL 是 http://192.168.1.108:5000/。

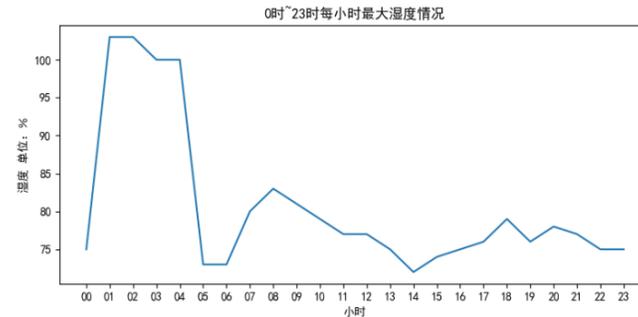
```
#导入 Flask 框架模块及其他相关模块，代码略
app = Flask(__name__)
@app.route('/')
def index():
    #从数据库读取温度和湿度数据,并返回页面,代码略
# 服务器其他功能，代码略
if __name__ == '__main__':
    app.run(host='192.168.1.108', port = 5000)
```

(4)请通过增加传感器和执行器对该系统功能进行一项扩展，写出增加的传感器和执行器名称及实现的功能 光线传感器测室内亮度 LED灯提示异常。或 气体传感器采集空气质量

(5)小华将系统中某天 24 小时的湿度数据导出，部分数据如第 14 题图 a 所示(时间格式为“时:分:秒”)，分析每小时的最高湿度值，线形图如第 14 题图 b 所示,部分 Python 序如下：

时间	类型	监测值
00:00:00	湿度	75
00:01:00	湿度	75
00:02:00	湿度	75
00:03:00	湿度	75
00:04:00	湿度	75
00:05:00	湿度	74
00:06:00	湿度	75
00:07:00	湿度	75
00:08:00	湿度	75

第 14 题图 a



第 14 题图 b

```
import pandas as pd
import matplotlib.pyplot as plt
dft = pd.read_csv('data.csv') # 读取文件 data.csv 中的数据
dft.insert(0, '小时', '') # 插入列
for i in dft.index:
    t = dft.at[i, '时间'] # 通过行标签和列标签选取单个值
    dft.at[i, '小时'] = t[0:2] 提取时间列的前两个数字存入‘小时’字段中，根据小时进行汇总
dfh = dft.groupby('小时', as_index=False).max() # 分组求最大值
plt.plot(dfh['小时'], dfh['监测值'])
#设置绘图参数，显示如第 14 题图 b 所示线形图，代码略
①请在程序中划线处填入合适的代码。
②小华分析线形图发现存在湿度值大于等于 100 的噪声数据，要删除 dft 对象中的噪声数据，下列代码段中，能正确实现的有 AD (多选, 填字母)。(注：全部选对的得2分，选对但不全的得 1分，不选或有选错的得0分)
A.dft = dft[dft['监测值']< 100]
B.dft = dft["监测值"]< 100 格式都不对了
C.n = len(dft[dft['监测值']>= 100])
dft = dft.sort_values('监测值') # 升序排序
dft = dft.tail(n) 留下>=100的n条 # 获取尾部数据行
D.for i in dft.index:
    if dft.at[i, '监测值']>= 100:
        dft = dft.drop(i) # 删除行
```

15.某工程包含 n 个任务(编号为 0~n-1), 每天可以有多个任务同时进行。某些任务之间有依赖关系, 如第 15 题图 a 所示, 任务 4 依赖于任务 1, 任务 1 依赖于任务 2, 即任务 2 完成后才可以开始任务 1, 任务 1 完成后才可以开始任务 4。不存在一个任务依赖于多个任务或多个任务依赖于同一个任务的情况。现已对该工程的依赖关系进行了梳理, 结果如第 15 题图 b 所示, 标记“T”表示依赖关系需保留, 标记“F”表示依赖关系需删除。根据每个任务完成所需的天数和梳理后的依赖关系, 编写程序, 首先删除标记为“F”的依赖关系, 然后计算工程最快完成所需的天数, 并以工程最快完成所需的天数为期限, 计算每个任务最晚必须开始的时间。



任务 A	任务 B	标记
0	5	T
5	4	F
4	1	T
1	2	T
2	3	F

注:任务 A 依赖于任务 B
第 15 题图 b

请回答下列问题:

(1)若某工程有 6 个任务,任务间依赖关系如第 15 题图 a 所示, 完成任务 0~5 所需天数分别为 2,1,3,5,1,6, 则工程最快完成需要 ▲8 天。

(2)定义如下 erase(lst)函数, 参数 lst 列表的每个元素表示一个依赖关系。函数的功能是删除标记为“F”的依赖关系, 返回保留的依赖关系的个数。

```
def erase(lst):
    i = 0
    j = len(lst)-1
    while i <= j:
        if lst[i][2] == 'T':
            i += 1
        else:
            if lst[j][2] == 'T':
                lst[i] = lst[j]
                i += 1
            j -= 1
    return i
```

以工程最快完成所需的天数为期限, 计算每个任务最晚必须开始的时间。工程最快完成所需的天数为 8。例如对于任务链 5→0, 任务 0 完成需要 2 天, 则至少需要从倒数第 2 天开始, 即顺数第 8-2+1=7 天开始, 对于任务 5 完成需要 6 天, 则至少需要从倒数第 2+6 天开始, 即顺数第 8-(2+6)+1=1 天开始。

若 i 条记录是 F, 则检索到最后一条 T 的记录, 把 T 的记录覆盖当前 i 条中的 F 记录

若 lst 列表依次存储第 15 题图 b 所示的依赖关系, 如 lst[0]为[0, 5, 'T'], 调用 erase(lst)函数, 则语句“lst[i] = lst[j]”的执行次数为 ▲1。

(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 proc(n, lst, task):
    pr = [0]*n
    w = [0]*n # w[i]存放任务 i 最晚必须开始的时间
    m = erase(lst)
    for i in ① range(m):
        task[lst[i][1]][1] = lst[i][0]
        pr[lst[i][0]] = 1
    c = []
    days = 0 # days 存放工程最快完成所需的天数
    for i in range(n):
        if pr[i] == 0:
            k = i
            s = 0
            while k != -1:
                c.append(k)
                s += task[k][0]
                k = task[k][1]
            if s > days:
                days = s
    for i in range(n-1, -1, -1):
        k = c[i]
        if task[k][1] == -1:
            w[k] = days - task[k][0] + 1
        else:
            w[k] = w[task[k][1]] - task[k][0]
    #输出 days, 以及保存在 w 中的每个任务最晚必须开始的时间, 代码略
    工程包含的任务数存入变量 n
    任务间的依赖关系存入 lst 列表
    lst[i]包含 3 项, 任务 lst[i][0]依赖于任务 lst[i][1], lst[i][2]存放保留/删除标记, 任务数据存入 task 列表
    task[i]包含 2 项, task[i][0]为完成任务 i 所需天数, task[i][1]的初值为-1
    代码略
    """
    proc(n, lst, task)
```

这一过程排除了孤立的任务, 重点在于把彼此依赖关系的任务的链接关系构造起来。lst 中已经删除了标记为“F”的依赖关系, erase(lst)函数返回值为 m, 正好代表了剩余的元素个数,

pr[i]==0, 表示 i 没有依赖, 即为任务链头

链表遍历, 此处出现的链表是 task, 将任务链中的任务加入 c, 以图 b 为例, c=[2, 1, 4, 3, 5, 0]

将任务链中的天数累加在一起, 求得最多天数
通过倒序遍历任务列表, 计算每个任务的最晚开始时间。对于每个任务编号 k, 如果该任务没有后续任务 (即 task[k][1]为-1), 则将最晚开始时间 w[k]设置为 days-task[k][0]+1, 表示该任务必须在工程最快完成时间内开始。否则, 将最晚开始时间 w[k]设置为其后续任务的最晚开始时间减去当前任务完成时间, 表示该任务必须在其后续任务完成前开始。

或 w[k]=w[c[i+1]]-task[k][0]