

绝密★考试结束前（向在家休养和隔离的莘莘学子致礼！你们辛苦了，预祝高考取得优异的成绩！）

Z20 名校联盟（浙江省名校新高考研究联盟）2023 届高三第二次联考

技术试题卷

信息命题：桐乡高级中学 朱海宁 审稿：海宁高级中学 朱晓薇 玉环中学 张炳岳 校稿：张红光、钱丹红
通用命题：萧山中学 徐奎 审稿：路桥中学 金霞 桐乡高级中学 王建献 校稿：毛尚、姚维红

考生须知：

1. 本卷满分 100 分，考试时间 90 分钟。
2. 答题前，在答题纸指定区域填写学校、班级、姓名、试场号、座位号及准考证号。
3. 所有答案必须写在答题纸上，写在试卷上无效；考试结束后，只需上交答题卷。

第一部分 信息技术（共 50 分）

一、选择题（本大题共 13 小题，每小题 2 分，共 26 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. 下列关于计算机硬件、软件和移动通信技术的说法，正确的是
A. windows10 功能强大，是一款优秀的应用软件
B. 学校图书馆的图书管理系统，包含多个模块，是一款功能完善的系统软件
C. 5G 移动通信网络，指的是数据传输速度达到了 5Gbit/s
D. 计算机硬件主要由控制器、运算器、存储器、输入设备和输出设备五大部件组成
2. 计算机的诞生为通过机器来模拟人类智能提供了无限的想象空间，促进了人工智能的发展。图灵为此进行了深入的思考，发表论文《计算机器和智能》，提出了著名的（ ），被人尊称为“人工智能之父”。
A. 图灵机 B. 信息论 C. 图灵测试 D. 人工智能这个名词

阅读下列材料，回答 3-5 题

智慧景区通过网络，全面、透彻和及时的感知景区的地理事物、自然资源、旅客行为、工作人员行迹、景区基础设施和服务设施的状态。

景区建设无线网络，实现 Wi-Fi 全覆盖，能逐步实现智慧景区建设，旅客可以充分体验景区的人性化服务。游客实现网络远程订票，凭身份证，实现景区“一卡通”。如果说在线旅游预订所服务的内容是游客的出行和住宿，那么智慧景区所要解决的则是游客到达景区后“最后一公里”的游览问题。

3. 下列关于智慧景区数据相关描述，不正确的是
A. 数据就是客观事物的符号表示
B. 数据在网络中传输，必须遵循 TCP/IP 协议
C. 对于景区的大数据而言，数据不再追求精确性，能够接受数据的混杂性。
D. 旅游旺季，该系统采用批处理方式对景区实时数据进行分析，对游客进行智慧导航
4. 为了数据安全，如果采用异或加密。用二进制编码为“00101100”的密钥 K 对明文进行简单异或加密，得到密文 C 的二进制编码是“01010101”，那么明文对应的二进制编码是
A. 00101001 B. 01111001 C. 01000100 D. 00111000

5. 下列对智慧景区系统描述不正确的是
- A. 身份证加入人脸识别, 提高了系统的安全性, 防止身份证被盗刷。
 - B. 身份证是有源电子标签
 - C. 出入口闸机故障, 会导致游客排队、引起拥堵, 说明信息系统对外部环境有依赖性
 - D. 游客直接使用身份证出入景点, 减少了工作人员, 提高工作效率
6. 未经压缩的 BMP 图像文件“a.bmp”, 其参数为 300*400 像素、256 色; 纯文本文件“b.txt”存储了 1 万个 GB2312 编码的汉字, 则文件“a.bmp”与“b.txt”存储容量之比约为
- A. 6:1
 - B. 12:1
 - C. 96:1
 - D. 3072:1
7. 下列 Python 表达式的值为 False 的是, 其中 s = “Python”, a, b 是两个任意整数
- A. “8” > “10”
 - B. s[5:3:-1] == “on”
 - C. int(-2.5) == -2
 - D. max(a, b) == (a+b+abs(a-b))//2
8. 链表不具备的特点是
- A. 所需存储空间与存储元素个数成正比
 - B. 插入、删除操作不需要移动元素
 - C. 无须事先估计存储空间的大小
 - D. 可随机访问任何一个元素
9. 采用冒泡排序算法, 对某数组数据进行排序, 经过一轮后的结果是“2, 3, 9, 5, 6, 7”, 那么下列说法不正确的是
- A. 这轮排序, 有可能没发生数据交换
 - B. 这轮排序, 有可能只发生了 1 次数据交换
 - C. 排序结束后, 数据是升序的
 - D. 完成全部排序后, 数据交换的次数和冒泡的方向无关
10. 有如下 Python 程序段
- ```
s = '1234567'
n = len(s)
k = -1
for i in range(n - 1):
 k = (k + 3) % len(s)
 s = s[k + 1:] + s[:k]
 k = -1
print(s)
```
- 执行改程序段后, 输出的结果是
- A. 1
  - B. 3
  - C. 4
  - D. 7
11. 一棵二叉树的前序遍历序列是 ABCDEFG, 后序遍历序列是 CBFEGDA, 则根节点的左子树节点的个数可能是
- A. 2
  - B. 3
  - C. 4
  - D. 5
12. 有如下 Python 程序段, 在无序的序列中利用对分查找算法求第 k 个大的数:
- ```
k = 3
a = [8, 4, 6, 7, 5, 6, 2, 4]
low, high = min(a), max(a)
```

```
while low <= high:
    m = (low + high) // 2
    cnt = 0
    for i in a:
        if i > m Then cnt += 1
    if cnt >= k:
        low = m + 1
    else:
        high = m - 1
```

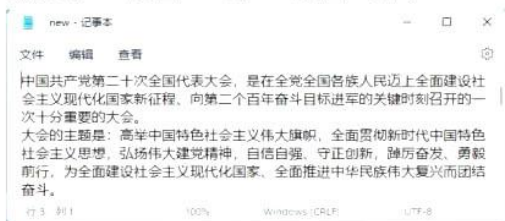
执行程序后，下列说法不正确的是

- A. 第 k 大数是 low
B. low > high
C. m 的值为 6
D. cnt 的值为 3
13. 国王将金币作为工资，发放给忠诚的骑士。第一天，骑士收到一枚金币；之后两天（第二天和第三天）里，每天收到两枚金币；之后三天（第四、五、六天）里，每天收到三枚金币；……这种工资发放模式会一直这样延续下去：当连续 N 天每天收到 N 枚金币后，骑士会在之后的连续 N+1 天里，每天收到 N+1 枚金币（N 为任意正整数）。现在骑士想知道，工作一定的天数后，他能获得的金币数量。小明写了 3 个 python 自定义函数，完成了骑士的愿望。这三个函数，从时间复杂度分析，程序效率从高到低依次是

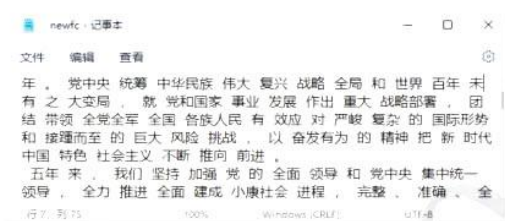
<pre>def f1(day): golds, i, s = 1, 0, 0 for j in range(day): s += golds i = (i + 1) % golds if i == 0: golds += 1 return s</pre>	<pre>def f2(day): golds, s = 1, 0 while day >= golds: s += golds * golds day -= golds golds += 1 s += golds * day return s</pre>
<pre>def f3(day): k = int((2 * day)**0.5 + 1e-6) if k * (k + 1) > 2 * day: k -= 1 s = k * (k + 1) * (2 * k + 1) // 6 + (day - k * (k + 1) // 2) * (k + 1) return s</pre>	

- A. f1>f2>f3
B. f2>f3>f1
C. f3>f2>f1
D. 三者相同
- 二、非选择题（本大题共 3 小题，第 14 小题 9 分，第 15 小题 8 分，第 16 小题 7 分，共 24 分）
14. 2022 年 10 月 16 日，习近平总书记在中国共产党第二十次全国代表大会上作了《高举中国特色社会主义伟大旗帜 为全面建设社会主义现代化国家而团结奋斗》的报告，大会后全文发表。现将该报告内容存储在“new.txt”文件中，经程序处理后的数据保存为“newfc.txt”文件。分

别如第 14 题图 a、第 14 题图 b 所示。



第 14 题图 a



第 14 题图 b

- (1) 处理文件“new.txt”中的数据生成“newfc.txt”文件的过程，一般称为：_____▲_____
- (2) 编写如下 python 程序，读取“newfc.txt”文件的数据，并统计其中各词语（2 个及以上汉字构成）出现的次数，在画线位置填写合适的代码，完善程序。

```
import pandas as pd
f = open('newfc.txt', 'r', encoding='utf-8').read() # 读取文件
words = f.split() # 利用空格、换行符将字符串 f 分割成列表
counts = {} # 建立空字典，用于存放词语及其出现的次数
for word in words: # 统计其中各词语（2 个及以上汉字构成）出现的次数
    if _____ ① _____ :
        if word not in counts:
            counts[word] = 0
            _____ ② _____
# 利用字典生成相应的 Series，根据值降序排列
s = pd.Series(_____ ③ _____).sort_values(ascending=False)
for i in _____ ④ _____ : # 从高到低输出出现次数最多的 10 个词语
    print(i)
```

15. 学校服务器机房设备贵重，要求保持恒温干燥的环境，且大部分时间处于无人值守的状态。管理人员搭建了智能监测系统，系统结构示意图如第 15 题图所示。智能终端通过传感器采集环境数据，通过 IoT 模块（物联网模块）传送相关数据到 Web 服务器并保存到相应数据库。Web 服务器将处理的结果经 IoT 模块传送给智能终端，再由智能终端启动执行器实现机房空调自动开启和关闭，同时管理人员还能随时远程监控机房环境状态。



- (1) 为了实现该功能，最合适的传感器是_____▲_____。（单选，填字母：A. 粉尘传感器/B. 光线传感器/C. 距离传感器/D. 温湿度传感器）
- (2) 根据该信息系统的功能要求，15 题图中的①②两处分别表示的设备名称是：_____▲_____，_____▲_____。（单选，填字母：A. IoT(物联网模块)；B. 路由器；C. 数据库；D. 智能终端）

- (3) 机房环境控制系统的相关代码如下：智能终端需要通过网络提交数据，IoT 模块(物联网模块)链接 wifi，部分代码如下：

```
IP="192.168.31.233"
PORT="8080"
SSID="zjtg"
PASSWORD="98765432"
Uart. init(baudrate=115200,bits=8,parity=None,stop=1,tx=pin2,rx=pin1)
While Obloq. connectWIFI(SSID,PASSWORD,10000)!=True:
    display. show(Image. SAD)
```

根据以上代码，物联网模块的 tx 针脚应接到智能模块扩展板的 ▲ 口。(单选，填字母 A. pin0 / B. pin1 / C. pin2)

- web 服务器用 Flask Web 框架实现简单网页应用程序，具体代码如下：

```
from flask import Flask, request
app = Flask(__name__)
@app.route('/')
def index():
    return 'index'
@app.route("/input", methods=['POST', 'GET'])
def add_data():
    sensorid = int(request.values.get('id'))
    sensorvalue = float(request.values.get('val'))
#其他代码略
if __name__ == '__main__':
    app. run(host='0.0.0.0', port=5000, debug=True)
```

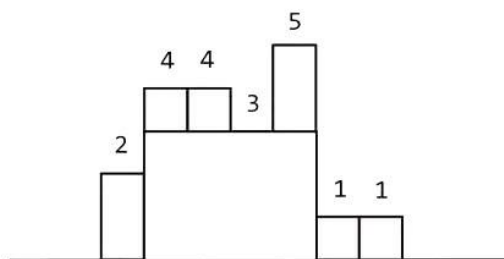
- (4) 下列关于 @app.route("/input", methods=['POST', 'GET']) 与 @app.route("/") 区别的说法，正确的是 ▲ (单选，填字母)

- A. 两个路由都只支持 'GET' 请求
- B. 两个路由都支持 'POST' 请求
- C. @app.route("/input", methods=['POST', 'GET']) 支持 'POST' 请求
- D. @app.route("/") 支持 'POST' 请求

- (5) 针对整个信息系统的搭建，下列说法正确的是 ▲ (多选，填字母)

- A. 搭建完成后系统测试包括硬件测试、软件测试、网络测试
- B. 使用 Flask Web 框架搭建的服务端程序，打开调试功能可以提高系统安全性
- C. 采用 C/S 模式或者 B/S 模式，属于开发模式选择
- D. 采用什么样的终端、具体的传感器型号选择，属于搭建前期的需求分析。

16. 一面木板墙，每块木板的宽度都是 1，现在想在木板墙上沿着平行于地面的方向切割出一块矩形区域。如果知道每块木板的高度，如何切出的矩形面积最大？木板墙如第 16 题图所示：



第 16 题图

图中有7块木板，每块木板的高度分别为：2、4、4、3、5、1、1。尝试后，我们发现最大的矩形就是波浪图案部分所示。也就是切割了高度为4、4、3、5四块木板，形成了一个高度为3，宽度为4的矩形，这个最大面积为12。

经过尝试，可以发现：切下来的最大矩形，一定是以最大矩形所在的区域中最短的那块木板为矩形的高度值。

这样我们可以枚举每一块木板，每次都以前木板作为高度。就是把当前木板当成是切出来的矩形区域中最矮的木板，然后分别向左右延伸，切出此时的最大矩形区域。当把所有的木板都尝试过一遍后，我们在所有的结果中比较出最大值，这个最大值就是我们要求的最大矩形面积。如果木板的数量为 n ，那么这种算法的时间复杂度接近于 $O(n^3)$ 。但是，如果利用好栈这种数据结构，就能快速的定位左右延伸的最远位置，将算法的时间复杂的降低到 $O(n)$ 。

实现上述时间复杂度为 $O(n)$ 的Python程序代码如下，完成以下问题。

(1) 如果7块木板从左往右依次的高度是2、1、4、5、1、3、3，切割出来的最大矩形面积是

_____▲_____。

(2) 完成划线处代码。

```
h = [2, 4, 4, 3, 5, 1, 1]
n = len(h)
# L[i]保存的是比当前木板高度低的且离当前木板最近的左侧木板位置
# R[i]保存的是比当前木板高度低的且离当前木板最近的右侧木板位置
L, R = [0] * n, [0] * n
st, top = [0] * n, -1
for i in range(n): # 生成L列表相应数据
    while top != -1 and _____ ① _____:
        top -= 1
    if top == -1:
        L[i] = -1
    else:
        L[i] = _____ ② _____
    top += 1
    st[top] = i
top = -1
for i in range(n - 1, -1, -1): # 生成R列表相应数据
    # 为防止泄露天机，此处代码隐藏
ans = 0
for i in range(n): # 遍历所有木板，取最大面积
    s = _____ ③ _____
    if s > ans:
        ans = s
print(ans)
```

关于我们

自主选拔在线是致力于提供新高考生涯规划、强基计划、综合评价、三位一体、学科竞赛等政策资讯的升学服务平台。总部坐落于北京，旗下拥有网站（[网址：www.zizzs.com](http://www.zizzs.com)）和微信公众平台等媒体矩阵，用户群体涵盖全国90%以上的重点中学师生及家长，在全国新高考、自主选拔领域首屈一指。

如需第一时间获取相关资讯及备考指南，请关注**自主选拔在线**官方微信号：**zizzsw**。



 微信搜一搜

 自主选拔在线

