2024 学年第二学期浙江省名校协作体试题

高二年级技术学科

考生须知：

1．本卷满分 100 分，考试时间 90 分钟；

2．答题前，在答题卷指定区域填写学校、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卷上，写在试卷上无效；

4．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 12 题，每题 2 分，共 24 分。每小题给出的四个选项中, 只有一个符合

题目要求)

阅读下列材料，回答第1至3题:

某医疗中心引入了智能健康监测系统，该系统通过可穿戴设备收集患者的生理数据，如心率、血压值和睡眠质量等实时监控患者的健康状况。通过文字、图像等形式呈现体质健康状况，进而优化治疗方案和提高患者护理质量。

1.下列关于数据、信息和知识的说法，不正确的是

A. 系统中的数据均是结构化数据

B. 可穿戴设备收集的血压值是数据的表现形式之一

C. 数据的客观性为医疗诊断提供了可靠的依据

D. 通过分析血糖水平数据得出糖尿病管理策略，这是知识的体现

2.下列关于大数据及数据处理的说法，正确的是

A. 系统收集的数据均为静态数据 B. 处理大数据时更加注重数据的相关性

C. 处理大数据一般采用并行思想 D. 系统分析的是全体数据，要求每个数据准确无误

3.下列关于保障患者数据安全的措施，合理的是

A. 定期备份患者数据 B. 保护数据的安全不需要保护存储数据的介质

C. 将患者数据随意分享给其他医疗机构 D. 将未经加密的患者数据存储在公共云服务器上

阅读下列材料，回答第4至5题:

智能健康监测系统还附有其他功能：电子病历功能，可存储患者的病历信息；疾病诊断辅助功能，可根据患者的症状和检查结果，给医生提供诊断建议；医疗影像分析功能，利用深度学习算法对医学影像进行自动识别和分析，准确识别病变等。

4.下列关于信息编码的说法，正确的是

A. 系统中的数据以二进制或十六进制形式存储

B. 医疗影像数字化的过程，一般需要经过采样、量化与编码

C. 电子病历用中英文结合来描述病情，每个字符均占1字节

D. 位深度为8位的数字化医疗影像，其量化值取值范围为0～256

5.下列关于人工智能在医疗领域的应用说法，正确的是  
A. 智能医疗辅助系统可以完全替代医生进行诊断和治疗  
B. 利用深度学习算法提升医疗水平，属于行为主义人工智能的应用  
C. 电子病历管理、疾病诊断辅助、医疗影像分析等均属于人工智能的应用  
D. 人工智能在医疗领域有很大的潜力，同时存在数据安全和隐私保护等问题

6.下列Python表达式中，值与其他三项不同的是

|  |  |
| --- | --- |
| A. len( [ 1 , 2 , 3 , 4 , 5 ] ) | B. ord( "g" ) - ord( "b" ) |
| C. int( max( "20" , "5" , "34" ) ) | D. str( abs ( 3 - 2 \*\* 3 ) ) |

7.正读和反读都一样的字符串称为“回文字符串”，现要判断某非空字符串s是否为“回文字符串”。实现该功能的程序段如下，划线处应填入的正确代码为

s = input("请输入字符串:")

n = len(s) ; k = n // 2

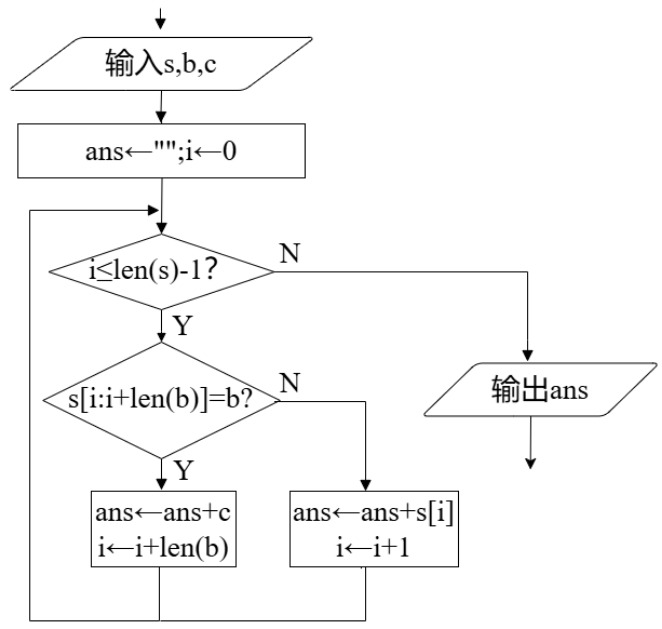
if \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_:

print("字符串:",s,"是回文字符串")

else:

print("字符串:",s,"不是回文字符串")

|  |  |
| --- | --- |
| A. s[ : k ] == s[ k + 1 : ][ : : -1] | B. s[ : k ] == s[ n - k : ][ : : -1] |
| C. s[ : k + 1] == s[ ( n + 1 ) // 2 : ][ : : -1] | D. s[ : k + 1] == s[ k + n % 2 : ][ : : -1] |



8.某算法的部分流程图如第 8 题图所示，若s的值为"good good study"，b 的值为 "good"，c 的值为 "hard"，执行这部分流程后，下列说法正确的是

A. 该流程图有 2 个分支结构

B. 该流程图循环了 15 次

C. 循环条件改成i≤ len(s) - len(b)不影响程序运行效果

D. 语句ans += c共执行了 2 次

9.图书馆读者的借阅状态会根据借阅次数进行更新。若读者在一个月内借阅次数少于 3 次（不含），借阅状态更新为 “不活跃”。如果借阅次数在 3 次至 8 次之间，

第8题图

借阅状态更新为 “活跃”。如果借阅次数超过 8 次（不含），

借阅状态更新为 “非常活跃”。根据算法设计程序，下列选项不正确的是

|  |  |
| --- | --- |
| A. status = "非常活跃"  if times <= 8:  if times < 3:  status = "不活跃"  else:  status = "活跃" | B. status = "活跃"  if times < 3:  status = "不活跃"  elif times > 8:  status = "非常活跃" |
| C.status = "非常活跃"  if times < 3 :  status = "不活跃"  if 3 <= times <= 8:  status = "活跃" | D. status = "不活跃"  if times > 8:  status = "非常活跃"  if times >= 3:  status = "活跃" |

10.有 Python 程序段如下：

n = 5 ; m = 6

a = [ [ i \* n + j for j in range(n) ] for i in range( m ) ] #生成了一个m行n列的二维列表

for i in range( m // 2 ):

for j in range(0 , n , 2) :

a[ i ][ j ] , a[ m - 1 - i ][ j ] = a[ m - 1 - i ][ j ] , a[ i ][ j ]

则执行程序后，a[ 0 ][ 0 ]和a[ 0 ][ 1 ]的值分别为

|  |  |  |  |
| --- | --- | --- | --- |
| 1. 25和1 | 1. 26 和 2 | 1. 0 和 1 | 1. 25 和 26 |

11.有 Python 程序如下：

def f(s):

n = len(s) ; i = 0 ; j = n - 1

ans = ""

while i < n // 2 and j >= n // 2 :

if s [ i ] >= s [ j ]:

ans += s [ i ]

j -= 1

else:

ans += s [ j ]

i += 1

return ans

s = input( "输入字符串: " )

print(f(s))

执行该程序段，分别输入下列选项中的字符串，输出结果不为“ppp”的是

|  |  |  |  |
| --- | --- | --- | --- |
| 1. python | 1. public | 1. purple | 1. phrase |

12.有 Python 程序如下：

import random

def encode(msg , key) :

result = ""

for i in range( 0 , len( msg )) :

c = msg[ i ]

if "a" <= c <= "z" :

result = chr( ( ord( c ) + key[ 0 ] - ord( "a" ) ) % 26 + ord( "a" ) ) + result

elif "A" <= c <= "Z":

result = result + chr( ( ord( c ) + key[ 1 ] - ord( "A" ) ) % 26 + ord( "A" ) )

elif "0" <= c <= "9" :

result = result + str( ( int( c ) + key[ 2 ] ) % 10 )

else:

result = c + result

return result

keys = [ -3 , -2 , -1 , 1 , 2 , 3 ]

msg = '8mA3-!'

key = random.sample( keys , 3 ) #随机抽取列表keys中的 3 项元素，得到一个新列表

re = encode( msg , key )

print( re )

程序运行后，输出结果可能是

|  |  |  |  |
| --- | --- | --- | --- |
| 1. !-k9Y6 | 1. !-6pB4 | 1. !-nY00 | 1. !-n7D2 |

二、非选择题（本大题共 3 小题，第13题8分，第14题9分，第15题9分，共26分）

13.亲和数是一对整数，其中每个数都是另一个数的真因数之和。真因数是指除了自身以外的所有因数，例如整数6的真因数有1,2,3。例如，220和284是一对亲和数，因为220的真因数之和是284，而284的真因数之和是220。以下Python程序用于在指定范围内查找所有亲和数对。程序运行结果如第13题图所示。请回答下列问题：

def divisors(n):

sum = 1

for i in range( 2 , int(n \*\* 0.5) + 1 ):

if n % i == 0:

sum += i

if ① :

sum += n // i

return sum

def find( start , end ):

pairs = []

for i in range( start , end + 1 ):

dsum = divisors( i )

if dsum > i and dsum != i and ② :

pairs.append( ③ )

return pairs

#主程序

start = int(input( ))

end = int(input( ))

第13题图

输入start：1

输入end：2000

输出：

亲和数对: 220 和 284

亲和数对: 1184 和 1210

pairs = find( start , end )

for pair in pairs:

print( "亲和数对: " , pair[ 0 ] , " 和" , pair[ 1 ] )

1. 请在划线处填入合适的代码。
2. 若删除程序加框处语句，输入start的值为1 , end的值为2000,将输出 对亲和数对。
3. “三位一体”是高校招生的一种选拔模式，其所依据的成绩主要由高考成绩、学考成绩和综合素质测试成绩按比例折算而成。现将某高校考生的学考成绩存储在“data.xlsx”文件中，部分数据如第14题图a所示。编写Python程序，实现将考生学考等级折算成相应的分数这一功能。

请回答下列问题：

|  |  |
| --- | --- |
| 109a934b75f12498a3f61001ed4031a | cb8332a8246d8eda12d73cec04ce668 |
| 第14题图a | 第14题图b |

1. 若某考生的学考成绩如表1所示,学考成绩折算方式如表2所示，则考生的折算分数为\_\_\_分。

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | 科目 | 语 | 数 | 英 | 物 | 化 | 生 | 历 | 地 | 政 | 技 | | 等级 | A | A | A | A | A | A | C | A | B | A | | |  |  |  |  |  | | --- | --- | --- | --- | --- | | 等级 | A | B | C | D | | 折算 | 10 | 9 | 7 | 4 | |
| 表1 | 表2 |

1. 统计学考成绩折算总分的Python程序如下：

import pandas as pd

df = pd.read\_excel( "data.xlsx" )

cj = { "A" : 10 , "B" : 9 , "C" : 7 , "D" : 4 }

for i in df.index:

s = 0

for j in df.columns[ 3 : 13 ] :

m = ①

if m in cj :

②

df.at[ i ,"总分" ] = s

①程序加框处可填入的代码有 (多选，填字母)（注：全部选对得2分，选对但不全得1分，不选或者有选错得0分）

|  |  |  |  |
| --- | --- | --- | --- |
| A.df.at[ i , j ] | B.df [ i ][ j ] | C.df [ j ][ i ] | D.df [ i , j ] |

②请在划线处填入合适的代码。

1. 学校想要了解学考成绩折算平均分居前3名的班级(若分数相同则一同输出)，并创建如第14题图b所示的图表，请在划线处填入合适的代码。

import matplotlib.pyplot as plt

df1 = df.groupby("班级" , as\_index = False).总分.mean( )

df2 = df1.sort\_values("总分" , ascending = False , ignore\_index = True) #ignore\_index是否重置索引

score = df2.at[2 , "总分"]

df3 = ①

plt.bar( ② ， ③ )

#图表设置，代码略

plt.show( )

15.小华开发了一个简单的医院门诊预约管理系统，用于模拟病人的预约和取消操作。系统输入格式如下:若输入"预约"开头，后面两个数字为预约开始时间和持续时长。若能找到符合病人预约要求的空闲时间段则预约成功，系统按照预约顺序分配病人编号，其中病人起始编号为1。若输入"取消"加病人编号，系统会查找指定编号并释放相应的预约时间段。

假设该门诊系统每天可预约的时间范围为0到99,则第15题图a门诊系统的时间段分配情况如第15题图b所示，每行表示一个时间段的预约情况。

|  |  |
| --- | --- |
| QQ_1733989133949 | QQ_1733989365378 |
| 第15题图a | 第15题图b |

请回答下列问题：

1. 若门诊的可预约时间为0~45，在执行了第15题图a的前4行输入后，门诊系统分配的时间段为： 0 - 9 : 空闲， 10 - 29 : 空闲， 30 - 39 : 病人2， 。（请完善最后一段时间段门诊预约情况）

(2) 实现上述功能的程序如下，请在划线处填入合适的代码。

(3) 程序中加框处代码有错，请改正。

#定义如下 book 函数，函数的功能是将病人分配到指定的时间段。

def book(sch, st, time, id):

ed = st + time - 1;i = 0

while i < len(sch):

block = sch[i]

if not block[1] and block[2] <= st and block[3] >= ed:

sch[i] = [id, True, st, ed]

if block[2] < st:

sch = sch[:i] + [ ① ] + sch[i:]

i += 1

if st < block[3] :

sch = sch[:i+1] + [[0, False, ed + 1, block[3]]] + sch[i+1:]

return sch, id

i += 1

return sch, -1

#定义如下 cancel 函数，函数的功能是找到并释放指定病人的预约时间段。

def cancel(sch, id):

    for block in sch:

        if ② and block[1]:

            block[0] = 0

            block[1] = False

return 0

    return -1

#主程序

'''sch列表用于存储每个时间段的预约情况，分别记录病人序号，是否预约，起始时间，结束时间。病人序号为0表示该时间段空闲。'''

sch = [ [0, False, 0, 99] ]  #假设该门诊系统每天可预约的时间为 0 到 99

np\_id = 1  # 下一个可分配的门诊编号

#将“appointments.txt”文件数据逐行添加到列表reqs中, 代码略

# reqs= [ ['预约','10','20'], ['预约','40','10'], ['取消','1'], ...... ]

for req in reqs:

if req[0] == "预约":

st, time = int(req[1]), int(req[2])

sch, res = book(sch, st, time, np\_id)

if res != -1:

③

elif req[0] == "取消":

c\_id = int(req[1])

cancel(sch, c\_id)

for block in sch:

if block[1]:

print( block[2], "-", block[3], ": 病人", block[0] )

else:

print( block[2], "-", block[3], ": 空闲" )