**Python编程考试基础知识部分样题**

假定在以下脚本执行前，已经执行了下面几条语句：

import numpy as np

import pandas as pd

一、单项选择题。选择你认为正确的输出结果。

1. In [1]: ‘3’\*5

Out [1]: （ ）

A. ‘555’ B. ‘33333’ C. 15 D. error

2. In [2]: 18>7, [1, 8]>[7]

Out [2]: （ ）

A. (True, True) B. (True, False)

C. (False, True) D. (False, False)

3. In [3]: (lambda x : x+2)(5)

Out [3]:（ ）

A. 5 B. 6 C. 7 D. error

4. In [4]: (1, 2) + (3, )

Out [4]: （ ）

A. (4, 2) B. (1, 5) C. (1, 2, 3) D. error

5. In [5]: S=’123456789’; S[-1:-4]

Out [5]: （ ）

A. ‘9876’ B. ‘876’ C. ‘678’ D. ‘ ’

6. In [6]: len(”123456\x65”)

Out [6]:（ ）

A.7 B. 8 C. 9 D. 10

7. In [7]: ’\t123\tgo\n’.strip()

Out [7]:（ ）

A. ‘\t123\tgo\n’ B. ‘\t123\tgo’ C. ‘123go’ D. ‘123\tgo’

8. In [8]: L=[1, 2, 3, 4, 5, 6, 7, 8, 9]; L[2], L[-2]

Out [8]: （ ）

A. (2, 7) B. (3, 7) C. (2, 8) D. (3, 8)

9. In [9]: [(1, 2, ‘3’), [4, 5]] > [(1, 2, ‘4’), [4, 6]]

Out [9]:（ ）

A. True B. False C. error D. None

10. In [10]: ’to be? or not to be?’.split()

Out [10]:（ ）

A. ['to', 'be', ‘or’, ‘not’, ‘to’, ‘be’] B. [’to be? or not to be?’]

C. ['to', 'be?', ‘or’, ‘not’, ‘to’, ‘be?’] D. ['to', 'be', ‘?’, ‘or’, ‘not’, ‘to’, ‘be’, ‘?’]

11. In [11]: bool(-2j)

Out [11]:（ ）

A.True B. False C. error D. -1

12. In [12]: x = ‘123’, ; len(x)

Out [12]:（ ）

A.1 B. 2 C. 3 D. 4

13. In [13]: all((1<2, None))

Out [13]:（ ）

A. True B. False C. (True, False) D. (False, False)

14. In [14]: d={3, 2, 1} - { 0, 1, 2}; 2 in d

Out [14]:（ ）

A. true B. false C. None D. False

15. In [15]: a, \*b, c=1, 2, 3, 4, 5, 6; print(max(b))

Out [15]:（ ）

A. 3 B. 4 C. 5 D. 6

16. In [16]: np.array([ 3, 4, 5 ]) + np.array(( 1, 2, 3 ))

Out [16]:（ ）

A. array([ 4, 6, 8]) B. array([ 3, 4, 5, 1, 2, 3])

C. array([ 1, 2, 3, 4, 5]) D. error

17. In [17]: np.arange(3)\*2

Out [17]:（ ）

A. array([ 0, 1, 2, 0, 1, 2]) B. array([ 0, 2, 4])

C. array([ 1, 2, 3, 1, 2, 3]) D. error

18. In [18]: arr1=np.linspace(0, 10, 6); arr1.shape

Out [18]:（ ）

A. (6, ) B. (, 6) C. (1, 6) D. (6, 1)

19. In [19]: np.eye(3) - np.array([0, 1, 0])

Out [19]:（ ）

A. array([[ 1., 0., 0.], [ -1., 0., -1.], [ 0., 0., 1.]])

B. array([[ 1., -1., 0.], [ 0., 0., 0.], [ 0., -1., 1.]])

C. array([[ 1., 0., 0.], [ 0., 1., 0.], [ 0., 0., 1.]])

D. array([[ 0., 1., 1.], [ 1., 0., 1.], [ 1., 1., 0.]])

20. In [20]: arr2d=np.arange(12).reshape(4,3); arr2d.T[::-1, 2]

Out [20]:（ ）

A. array([11, 8, 5, 2])

B. array([10, 7, 4, 1])

C. array([5, 4, 3])

D. array([8, 7, 6])

二、辨析题（每题2分，共20分）每题仅有一个正确的答案。

21. 下面哪个名字可以作为变量名使用：（ ）

A. def B. 0x86 C. Class D. it’s

22. In [1]: score = {‘jack’ : 90, ‘mike’ : 85}

下面哪个语句不会导致错误：（ ）

A. score[‘JACK’] B. score[85] C. score.get(‘tom’, 0) D. del score[jack]

23. 在notebook中显示当前工作目录的魔术命令为：（ ）

A. %pwd B. %path C. %home D. %hist

24．数据类型np.float64指的是（ ）

A. 8位浮点数 B. 4字节浮点数 C. 8字节浮点数 D. 8字节复数

25.下列哪个对象是不可变对象：（ ）

A. {2, 3, 4} B. [2, 3, 4] C. (2, 3, 4) D. [2, 3, (3, 4)]

26. 函数adding定义为：

def adding(a,b=0,c=None):

pass

请问下面的调用方式哪种是正确的？（ ）

A. adding(c = 5, 3) B. adding(b=5, a=3)

C. adding(c = 5) D. adding(b=5)

27. 在不引入任何扩展包时，下列哪个语句不能正常执行：（ ）

A. a1 = round(3.25, 2)

B. a2 = int(3.25)

C. a3 = round(3.25)

D. a4 = log(3.25)

28. 使用pyplot作图时，线型参数’g--x‘对应的曲线线型为：（ ）

A. 蓝色实线，不标记数据点 B. 蓝色虚线，数据点以’x’标记

C. 绿色实线，不标记数据点 D. 绿色虚线，数据点以’x’标记

29. 关于DataFrame对象的fillna方法，错误的说法是：（ ）

A. 若参数为value=1，则缺失值使用数值1填充；

B. 若参数为method = ‘ffill’，则使用缺失值记录之前的有效值填充；

C. 若参数为method = ‘bfill’，则使用缺失值记录之前的有效值填充；

D. 若参数为method = ‘pad’，则使用缺失值记录之前的有效值填充；

**pad/ffill**：用**前一个**非缺失值去填充该缺失值

**backfill/bfill**：用**下一个**非缺失值填充该缺失值

30. 下面关于DataFrame对象的merge方法，错误的说法是：（ ）

A. 参数left\_on指定左表用作连接键的列名；

B. 参数on指定用作连接键的列名；

C. 参数left\_index指定左表连接是否使用行索引；

D. 内连接和外连接生成的表长度一定不相同。

三、程序分析题。

31. 写出以下代码的运行结果与S的简要计算过程。形式如：结果为10 ( S= 1+2+3+4)。

S=0

for i in range(100):

if i<5: S += 1

elif 5<i<99: continue

elif 90<i<100: S +=2

else: S += 3

print(S)

32. 写出以下代码的运行结果与T的简要计算过程。形式如：结果为4 ( T= 1+1+2)。

L=[3, 1, 4, 1, 5, 0, 0.5]

T = 0

for i in L:

if i: T += 1

else: continue

if i%2 == 0: T +=2

print(T)

33. 请分别简要指出语句[1]-[3]的结果。

class light:

cnt = 1

def \_\_init\_\_(self, name):

self.name = name

def show(self):

print(self.cnt)

def tell(self):

print(light.cnt)

t1 = light('geek')

t1.cnt = t1.cnt + 5 # [1]

t1.show() # [2]

t1.tell() # [3]

33. 补充完整下面的类代码，使得当使用Man的实例对象m设置\_\_age属性时，即使用类似m.age = 20.21的代码时，会自动对年龄向下取整。当传递的属性值不是数值类型时，提示属性值非法，不对\_\_age属性进行修改。

class Man:

def \_\_init\_\_(self, age):

self.\_\_age = int(age)

@property

def age(self):

return self.\_\_age

@age.setter

def age(self, age):

\_\_\_\_\_\_\_\_#[1]

\_\_\_\_\_\_\_\_#[2]

* 以上题目仅用于展示可能的题型，不代表最终考试的范围和考点。
* 友情提示：考试卷中有一些坑，注意别跳进去，例如第22题和第31题。不要当无辜的小绵羊。
* 关于题目的难度和比例分配。以列表元素选择和列表切片操作为例：

a = [1, 2, 3, 4, 5, 6]

a[1] = ? 难度 \*

a[-1] = ? 难度 \*\*

a[1:3] = ? 难度 \*\*

a[-3:-1] = ? 难度 \*\*

a[3:1] = ? 难度 \*\*\*

a[-1:-3] = ? 难度 \*\*\*

a[-10:-3] = ? 难度 \*\*\*\*

逻辑推理顺序：

1. 知道a[-1]或a[1:3]的同学一定知道a[1]的含义，反之不成立，所以后者的难度高一级

2. 知道a[-1:-3]的同学肯定知道a[-3:-1]的含义，反之并不成立，所以难度高一级

3. 知道a[-10:-3]的同学肯定知道a[-3:-1]和a[-1:-3]的含义，所以难度高一级

题目难度比例（大概）：

难度\* 10%左右

难度\*\* 40%左右

难度\*\*\* 30%左右

难度\*\*\*\* 20%左右