

分数：100.00 题数：38

A、apple(n)+2

C、apple(n-1)+2

D、apple(n+1)-2

难度：一般

```

    &nbsp; &nbsp; &nbsp; &nbsp; else:

```

```
print(fib(m))
```

D、语句“def fib(n):”中的 n 为形参

难度：较难

[illegible]

D、函数的形参在函数被调用时获得初始值

试题编号：20210129-TR-004

题型：单选题

答案：D

难度：一般

试题解析：函数的形参作用域为本函数，在函数被调用时获得初始值。

6. 关于递归与递推方法的比较，错误的观点是？（ ）

A、递归是将复杂问题降解成若干个子问题，依次降解，求出低阶规模的解，代入高阶问题中，直至求出原问题的解；

B、递推是构造低阶的问题，并求出解，依次推导出高阶的问题以及解，直至求出问题的解；

C、数学上的递推关系可以通过递归的方法来实现；

D、递归算法代码简洁，运行速度比递推快，因此应该尽量采用递归的方法；

试题编号：20210206-xm-024

题型：单选题

答案：D

难度：一般

试题解析：D.递归的运行开销大。

7. 运行以下代码，输出结果正确的是？（ ）

```
a=1
b=c=[]
def fun(a,c):
    &nbsp; &nbsp; &nbsp; a=2
    &nbsp; &nbsp; &nbsp; c.append(a)
    fun(a,c)
print(a,b,c)
```

A、2 [2] [2]

B、1 [] [2]

C、1 [2] [2]

D、2 [] [2]

试题编号：20210129-TR-011

题型：单选题

答案：C

难度：较难

试题解析：函数体内变量 a 的值为 2，添加到了列表 c 中，b 和 c 指向同一个列表地址，因此列表 b 也随之改变

8. 关于 Turtle 库的表述中，错误的是？（ ）

A、Turtle 库是 Python 语言中一个很流行的绘制图像的函数库。

B、画布就是 turtle 为我们展开用于绘图区域，我们可以设置它的大小和初始位置。

C、turtle.circle()是只能画一个指定半径为 r 的圆。

D、turtle.speed(speed)：设置画笔移动速度，画笔绘制的速度范围[0,10]整数，数字越大越快。

试题编号：20210128-ph-018

题型：单选题

答案：C

难度：一般

试题解析：turtle.circle()是 turtle 中的常用命令，基本语法是 circle(radius,e)，即画一个指定半径为 r，角度 e 的圆或弧

9. 有 100 枚金币，其中有 1 枚轻 1 克的假金币，现在要找出这枚假金币，但身边只有 1 个没有刻度的天秤。小明先是将金币分成 50 枚一堆，共两堆称重，在轻的那一堆中又分成两堆，接着在轻的 25 枚中分成 12,12,1 三堆称重，若两堆 12 枚的重量相同，则假币为单独剩下的那一枚，否则在轻的那一堆中继续按照之前的办法称下去，直到找到假金币。请问小明采用的办法与哪个算法有着相似之处？

()

A、递归

B、分治

C、枚举

D、贪心

试题编号：20210130-lfy-029

题型：单选题

答案：B

难度：容易

试题解析：分治算法就是对一个问题采取各个击破的方法，将一个规模为 N 的问题分解为 K 个规模较小的子问题，这些子问题相互独立且与原问题性质相同。只要求出子问题的解，就可得到原问题的解。

10. 运行以下代码，正确的打印结果是？ ()

```
def f(s):
    &nbsp; &nbsp; &nbsp; &nbsp; t=0
    &nbsp; &nbsp; &nbsp; &nbsp; max=0
    &nbsp; &nbsp; &nbsp; &nbsp; for i in s:
    &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; if i>="0" and i<="9":
    &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; t=t+1
    &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; else:
    &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; if t>max:
    &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; max=t
    &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; t=0
    &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; print(max)
list="123ab45cd6d"
f(list)
```

A、 0

B、 1

C、 2

D、 3

试题编号：20210206-lfy-007

题型：单选题

答案：D

难度：一般

试题解析：本段代码中，函数 f()的作用是求最长的连续数字字符串的长度。

11. 下列关于函数的描述正确的是？ ()

A、函数是可重复使用的，用来实现单一，或相关联功能的代码段

试题解析：

难度：一般

试题解析：

```

    &nbsp; &nbsp; &nbsp; if x==1:

```

A、 120
B、 24120
C、 144
D、 12024

试题解析：由于 $\text{fact}(5)=\text{fact}(4)*5$, $\text{fact}(4)=\text{fact}(3)*4$, $\text{fact}(3)=\text{fact}(2)*3$, $\text{fact}(2)=\text{fact}(1)*2$, $\text{fact}(1)=1$, 所以 $\text{fact}(5)+\text{fact}(4)=120+24=144$ 。

D、 result= lambda x,y: x if x> y else y

试题解析：匿名函数的定义语法：lambda 参数：表达式语法。先写 lambda 关键字，然后依次写匿名函数的参数，多个参数中间用逗号连接，然后是一个冒号，冒号后面写返回的表达式。

A、 4
B、 8
C、 -8
D、 2

答案：B

试题解析：自定义函数 fun(m,n)的作用是求 m,n 两个整数的最大公约数。

```
def fun(x,y=5):
    return x*y
a=fun(10,10)
print(a)
```

- 试题编号：20210129-TR-012

难度：一般

- A、 Local
- B、 nonlocal
- C、 global
- D、 nonglobal

难度：较难

A、 op
B、 op(x,y,z)
C、 op x,y
D、 op(x+y,z)

难度：一般

```
A、 if n%2==0:
```

```
         x=x*2
```

```
else:  
    &nbsp; &nbsp; &nbsp; x=x+2
```

B、 n=1
while n<=100:
 x=x+2

C、 for i in range(100):
 for j in range(10):
 x=x+2

D、 for i in range(100):
 for j in range(i):
 x=x+2

试题编号：20210129-TR-031

题型：单选题

答案：A

难度：较难

试题解析：时间复杂度按数量级递增顺序为：常数阶、对数阶、线性阶、线性对数阶、平方阶

21. 下列哪个不是 Python 中的内建函数？（ ）

A、 asc(x)

B、 ord(x)

C、 chr(x)

D、 abs(x)

试题编号：20210129-lfy-005

题型：单选题

答案：A

难度：容易

试题解析：Python 中没有 asc(x)这个内建函数，ord(x)函数的作用是返回 x 对应的 ASCII 值，chr(x)函数的作用是返回 x 对应的字符，abs(x)函数的作用是返回 x 的绝对值。

22. 关于函数的定义语句，以下几项中正确的是？（ ）

A、 def f(c=2,a,b):

B、 def f(a,b=2,c):

C、 def f(*c,**d,a,b):

D、 def f(a,b,*c,**d):

试题编号：20210129-TR-014

题型：单选题

答案：D

难度：一般

试题解析：顺序实参必须要放在关键字实参之前，位置参数*args 必须要在关键字参数**kwargs 前，否则会程序报错。

23. 下列关于递归的描述不正确的是？（ ）

A、 递归函数一定包含条件控制语句

试题解析：递归算法一般代码简洁，易于理解，但执行效率较低，空间复杂度较高。

24. 下列哪个不是 Python 第三方库的 pip 安装方法？（ ）

试题解析：第三方库安装

25. 对于下列递归式子，当 $n=4$ 时，F 的值是？（ ）

$$F(n) = F(n-1) + 3 \quad \text{and} \quad F(1) = 2$$

试题解析：用递归式子推导得出。

二、判断题 (共10题, 每题2分, 共20分)

26. def add(a, b, c=0):
 return a+b+c
print(add(1, 2, 4))

这段程序的运行结果为 3。

试题编号：20210129-TR-008

题型：判断题

答案：错误

难度：一般

试题解析：在调用函数时给函数提供了实参时，Python 将使用指定的实参值

27. 算法复杂度分析的目的是分析算法的效率，以求改进。

试题编号：20210131-TR-032

试题解析：

运行以上程序，输出结果是 15。

试题解析：

试题解析：

试题解析：

试题解析：当全局变量和局部变量同名时，首先会找程序内部有没有局部变量，如果有，则调用，如果没有，才会去调用全局变量。

```
print(lst)
```

试题解析：Python 中字母区分大小写，所以自定义函数创建和调用时，函数名要一致，否则调用不成功。

一个人开始报数，数到第 m 个人又出局，...，如此反复到只剩下最后一个是胜利者。设 n 个人的编号分别为 $1, 2, \dots, n$ ，打印出局的顺序。

根据上述算法思想，补全下列代码。

输入输出示例：当 $n=10$ ， $m=4$ ，输出如下：

出局的人是：4

出局的人是：8

出局的人是: 2

出局的人是：7

出局的人是: 3

出局的人是: 10

出局的人是：9

出局的人是: 1

出局的人是: 6

最后胜利者是: 5

```
def fun(n,k):
```

```
L = list( )
```

```

    if n == 1:

```

[illegible][illegible]
$$x = 0$$

```

for i in

```

 ②

x = 3 - 1

[illegible]
$$\frac{1}{\det L[x]}$$

```

if x < 0:

```

$$x = 0$$
[illegible]

 ④

```
fun(10,4)
```

试题编号：20210128-ph-036

题型：编程题

答案：

评分标准： ① range(1,n+1) 或等效答案；（3分）

② range(n-1): 或等效答案; (3分)

③ $(x + k) \% \text{len}(L)$ 或等效答案； (3分)

④ L[0] 或等效答案； (3 分)

难度：较难

试题解析：

参考程序： def fun(n,k):

```

L = list(range(1,n+1))

```

```
    if n == 1:
```

```
      return
```

```

    &nbsp; &nbsp; &nbsp; else:

```

$$\text{ } \text{ } \text{ } \text{ } \text{ } x = 0$$

```
        for i in range(n-1):
```

```

x = (x + k) % len(L) - 1

```

```
print('出局的人是:',L[x])
```

```

del L[x]

```

```

    if x < 0:

```

$$x = 0$$

```

    &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; print('最后胜利者是:',L[0])

```

fun(10,4)

38.

设计一个算法，将一个正整数分解质因数。

程序分析：对 n 进行分解质因数，应先找到一个最小的质数 k ，然后按下述步骤完成：

这个质数恰等于 n ，则说明分解质因数的过程已经结束，输出即可。

 (2)如果 $n > k$, 但 n 能被 k 整除, 则应打印出 k 的值, 并用 n 除以 k 的商, 作为新的正整数 n , 重复执行第一步。

 (3)如果 n 不能被 k 整除，则用 $k+1$ 作为 k 的值，重复执行第一步。

根据上述算法思想，补全下列代码。

输入输出示例：当 $n=105$ ，输出： $105=3*5*7$

当 $n=60$ ，输出： $60=2*2*3*5$

```
def fun(n):
```

[illegible]

```
     for i in <range(0, len(arr)):
```

```
while n!=i:
```

```

if n>i and ②:

```



```

    if n>i and n%i==0:

```

```

    print(i,end='*')

```

```

    n=n//i

```

```

else:

```

```

    break
    else:

```

```

    print(n)

```

```

break

```

```

    if __name__ == "__main__":
        while True:
            num=input("输入一个正整数 : ")
            if not num.isdigit():

```

```

        break
        fun(int(num))

```