(2)

Manh Cong NGUYEN (manhcuongeic@gmail.com)

Campagne: [SG] Python Campaign 2018 - #2 Langage(s) de programmation: Python3 Langage: Anglais Date: 06/11/2018

SCORE

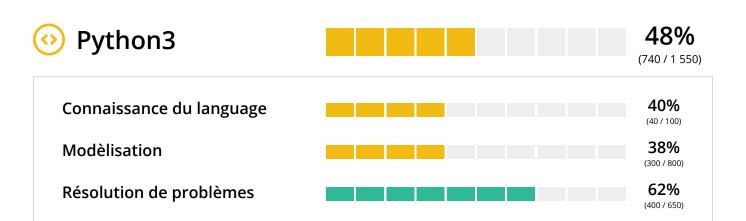
48% 740 / 1 550 pts

RANG

DURÉE

0H41

/ UH46





Question 1: Iterate through dictionary



Python3



00:25 / 00:45 0 / 20 pts





'd' is a dictionary, '1' is a list. Which code is incorrect?



\bigcap for k, v in \bigcirc	d.items():
----------------------------------	------------

for i, v in enumerate(l):

for k, v in d:

for k in d:

Résultat

Réponse incorrecte Connaissance du language +20pts



Question 2: Strings mutability



Python3



00:09 / 00:20



0/20 pts



Python strings are mutable



Réponse



False



True



Réponse incorrecte
Connaissance du language #20pts



Question 3: List slices





Python3 () 00:25 / 00:25



0 / 20 pts



What is the output?

1 = [1, 2, 3, 4, 5, 6]print(1[5:3:-1])

Réponse



b. [6, 5, 4]

c. [4, 3, 2]

d. [5, 4, 3]

Résultat

Réponse incorrecte
Connaissance du language +20pts



Question 4: Classmethod



Python3



00:27 / 01:30



20 / 20 pts



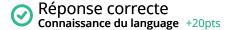
The @classmethod decorator alternates a method so that...

🕜 Réponse

	it could	be called	with classes	as arguments
--	----------	-----------	--------------	--------------

...when invoked as an object method, the calling object is ignored

...it becomes a class-level method, that is impossible to invoke as an object method





Question 5: Tuple assign



Python3



00:06 / 00:25 20 / 20 pts



Question

What is the output?

a, b = 1, 2

b, a = a, b

x = a, b

print(x)

Réponse

- 2, 1,
- (1, 2)
- **?** (2, 1)

Résultat

Réponse correcte Connaissance du language +20pts



Question 6: "Invert" a dictionary



Python3



02:00 / 02:00



0 / 50 pts

Le temps alloué à cette question s'est écoulé. La réponse du candidat a été automatiquement récupérée à la fin du décompte.

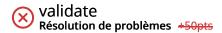
Question

Implement a Python function that "inverts" a dictionary where keys are strings, and values are integers.

So that dictionary {'k1': v1, 'k2': v2} would become {v1: 'k1', v2: 'k2'}.

Réponse

```
# Python code below
2 # Use print("messages...") to debug your solution.
4 def invert_dict(d):
     # Your code goes here
      k={ }
     for i,j in d.items:
        k[j]=i
    inverted_d=k
     return inverted_d
```





Question 7: Vector multiplication



Python3



01:34 / 03:00



100 / 100 pts



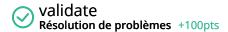
Implement a function for vector multiplication. Vectors are represented as Python lists.

(Vector multiplication: sum of the multiplication of individual elements at identical indexes.)



Réponse

```
1 # Python code below
2 # Use print("messages...") to debug your solution.
4 def vect_mul(v1, v2):
    # Your code goes here
    result=0
    for i in range(len(v1)):
       result+=v1[i]*v2[i]
    return result
```





Question 8: Animal sounds



Python3



06:29 / 07:00



300 / 300 pts



Given an input string like:

"Well, sheep says beeeeee and cat says miaaaaaaaaaaaa - and a cow would shout mooooooooow".

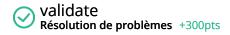
Extract the animal sounds from the string into a list.

(Words that include character repetions longer than a 2-length sequence, as shown in the example.)



Réponse

```
1 # Python code below
2 # Use print("messages...") to debug your solution.
3 def get_words(s):
       return s.split()
5 def find_sound(word):
     d={ }
      for i in word:
       d[i]=d.get(i,0)+1
8
9
          if d[i]>len(word)/2 and len(word)>2:
10
              return True
11
     return False
12
13 def animal_sound(s):
   # Your code goes here sounds=[]
15
16
     words=get_words(s)
17
     for word in words:
       if find_sound(word):
18
19
              sounds.append(word)
2.0
     return sounds
```





Question 9: Reverse poem line



Python3



06:00 / 06:00



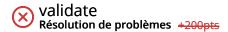
Le temps alloué à cette question s'est écoulé. La réponse du candidat a été automatiquement récupérée à la fin du décompte.

Question

Write a function that takes a poem and a line number as input, and prints a specified line reversed. See the format of a poem in the example code (ascii string with '\n' at the end of the lines).

Réponse

```
1 # Python code below
 2 # Use print("messages...") to debug your solution.
 3 def split_poem(poem):
      return poem.split('\n')
5 def reverse_line(st):
     s=" "
      for i,c in enumerate(st):
          s + = st[len(st) - i]
      return s
10 def reverse_line(poem, n):
# Your code goes here
      lines=split_poem(poem)
12
     #print(lines[n-1])
#result=""
14
     result=reverse_line(lines[n-1])
     return result
```





Question 10: Sorted list class



Python3



07:00 / 07:00 0 / 200 pts



Le temps alloué à cette question s'est écoulé. La réponse du candidat a été automatiquement récupérée à la fin du décompte.

Question

Implement a class that represents sorted lists (of basic orderable Python types, such as ascii characters or integers).

The test code defines expected behavior.

Réponse

```
1 # Python code below
2 # Use print("messages...") to debug your solution.
4 # Implement the SortedList class
6 class SortedList:
    def __init__(self, 1):
          self.l=l.sort()
   def add(self,item):
       self.l.append(item)
10
11
         self.1.sort()
    def concat(self, s2):
     self.1.extend(s2.1)
         self.l.sort()
```





Question 11: Binary search tree



Python3



09:00 / 09:00



0 / 300 pts

Le temps alloué à cette question s'est écoulé. La réponse du candidat a été automatiquement récupérée à la fin du décompte.

Question

Implement a binary tree that is binary search tree. (Left descendants of a node are smaller, right descendants of a node are bigger or equal than the value held in the node.). Extending the tree has to keep this property.

Complete the sample code so that it correspons to the test code.

Réponse

```
1 # Python code below
 2 # Complete the definition of the class
4 class Node:
      def __init__(self, n):
         self.value = n
          self.left = None
9
          self.right = None
10
     def add(self, n):
11
12
        if sel
13
          if self.value<n:
14
              add(self.right,n)
15
          else:
16
              add(self.left,n)
17
     def find(self, n):
19
       if self.left==None and self.right==None and self.value!=n:
              return False
22
         if self.value==n:
23
              return True
24
          if self.value<n:</pre>
              return find(self.right,n)
26
              return find(self.left, n)
```







Question 12: Logfile



Python3



07:25 / 08:00



300 / 300 pts

Question

We have a log file of the following format (separated by dash):

```
<timestamp>-<severity>-<message>
```

We would like to process it per record represented as a dictionary of the following format:

{'timestamp': '<timestamp>', 'severity': '<severity>', 'message': '<message>'}

What could be a most convenient solution?





```
1 # Python code below
 2 # Use print("messages...") to debug your solution.
4 # code for 'processlog' goes below
5 def get_lines(logs):
     return logs.split('\n')
8 def split_line(line):
    return line.split('-')
10
11 logs=""dkjkqjkd
12 qdkqjdfk
13 sqkjdkjqk
15
16 print (get_lines(logs))
17 def processlog(logs):
18
      lines=get_lines(logs)
     d=[]
20
     for line in lines:
        d_t={}
21
22
        timestamp.severity.message=split_line(line)
        d_t['timestamp']=timestamp
23
        d_t['severity'] = severity
      d_t['message']=message
d.append(d_t)
25
    return d
27
```





Glossaire

Connaissance du langage

La mesure de cette compétence permet de déterminer l'expérience du candidat dans la pratique d'un langage de programmation. Privilégiez cette compétence si, par exemple, vous recherchez un développeur qui devra être rapidement opérationnel.

Design

Cette mesure fournit une indication sur la capacité du candidat à appliquer des solutions standard pour résoudre des problèmes récurrents. Un développeur ayant un bon niveau dans cette compétence augmentera la qualité (maintenabilité, évolutivité) de vos applications. Cette compétence ne dépend pas spécifiquement d'une technologie. Privilégiez cette compétence si, par exemple, vous recherchez un développeur qui sera amené à travailler sur les briques qui structurent vos applications, à anticiper les besoins de demain pour développer des solutions pérennes.

Résolution de problèmes

Cette compétence correspond aux aptitudes du candidat à comprendre et à structurer son raisonnement pour trouver des solutions à des problèmes complexes. Cette compétence ne dépend pas spécifiquement d'une technologie. Privilégiez cette compétence si, par exemple, vos applications ont une composante technique importante (R&D, innovation).

Fiahilité

La fiabilité caractérise la capacité du candidat à réaliser des solutions qui prennent en compte les cas particuliers. Plus cette compétence est élevée, plus vos applications sont robustes (moins de bugs).

