

IC1T - Exemple d'examen

Question 1

Que vaut **a** à la fin de l'exécution du code suivant :

```
1  a = 3 ** 2
2  a = a // 2
```

- | | | | | |
|------|--------|-------|--------|------|
| a) 3 | b) 2 | c) 6 | d) 9 | e) 5 |
| f) 4 | g) 4.5 | h) 16 | i) 2.5 | j) 8 |

Question 2

Que va afficher le code suivant :

```
1  def fun(li, value):
2      res = []
3      for item in li:
4          if item > value:
5              res.append(value)
6          else:
7              res.append(item)
8      return res
9
10 print(fun([2, 6, 9, 1, 3, 8], 5))
```

- | | | |
|-----------------------|-----------------------|--------------------------|
| a) [] | b) [5, 6, 9, 5, 5, 8] | c) [2, 6, 9, 1, 3, 8] |
| d) [1, 2, 3, 6, 8, 9] | e) [2, 1, 3] | f) [2, 6, 9, 1, 3, 8, 5] |
| g) [5, 5, 5, 5, 5, 5] | h) [2, 5, 5, 1, 3, 5] | i) [6, 9, 8] |

Question 3

La fonction suivante cherche la valeur `value` dans la liste `L` et retourne la liste des indices où `value` a été trouvée. Par quelle instruction faut-il remplacer le commentaire de la ligne 4 ?

```
1 def findAll(L, value):
2     res = []
3     for i, item in enumerate(L):
4         # instruction manquante
5         res.append(i)
6     return res
```

- | | | |
|-----------------------------------|--------------------------------------|-----------------------------------|
| a) <code>for i in item:</code> | b) <code>if item == value:</code> | c) <code>for i in value:</code> |
| d) <code>if item = value:</code> | e) <code>while item == value:</code> | f) <code>while value in L:</code> |
| g) <code>while i == value:</code> | h) <code>if value in L:</code> | i) <code>if item in L:</code> |

Question 4

La fonction `split()` dans le code qui suit devrait diviser une phrase en une liste de mots

```
1 def split(sentence):
2     words = []
3     word = ''
4     for char in sentence:
5         if char == ' ':
6             words.append(word)
7             word = ''
8         else:
9             word += char
10
11     return words
```

Seulement, si on utilise la fonction comme ceci :

```
print(split("Je suis un as de la programmation"))
```

On obtient :

```
['Je', 'suis', 'un', 'as', 'de', 'la']
```

Que faut-il changer pour corriger la fonction ? (*les → indiquent le niveau d'indentation*)

- a) ligne 2 : → `words = [sentence]`
- b) ligne 3 : → `word = sentence[0]`
- c) ligne 4 : → `for char in range(len(sentence)):`
- d) ligne 5 : → → `if char == word[0]:`
- e) ligne 6 : → → → `words.append(char)`
- f) ligne 7 : → → → `word = sentence[0]`
- g) ligne 9 : → → → `word.append(char)`
- h) ligne 10 : → `words.append(word)`
- i) ligne 11 : → `return word`