

Commencé le jeudi 20 octobre 2022, 18:00

État Terminé

Terminé le jeudi 20 octobre 2022, 19:07

Temps mis 1 heure 7 min

Note 19,00 sur 20,00 (95%)

Description

L'examen dure 1h50 et se ferme automatiquement à la fin du temps. L'examen comporte 10 questions et vaut pour 20% de la note globale.

- Le professeur ne répondra à aucune question durant cet examen ;
- Toute communication entre étudiants est interdite ;
- Veuillez lire attentivement les instructions de chaque question ;
- L'indentation des programmes doit être faite correctement et ils doivent fonctionner sur Moodle ;
- Vérifier bien vos réponses avant de soumettre votre examen.

Si vous avez une hésitation sur la compréhension des questions, qui vous empêche d'y répondre, veuillez l'indiquer dans la case de note sur cette page ou en commentaire pour les questions de code.

Question 1

Terminé

Non noté

Sur mon honneur, j' compléter cet examen par moi-même, sans communication avec d'autres personnes.

Afin de ne pas créer de situations inégales pour les futures cohortes, j' ne pas copier et partager le contenu de cet examen.

Une fois ma note obtenue et mes réclamations faites et traitées, j' supprimer tout le matériel que j'ai créé ou copié afin de répondre aux questions dans les 30 jours qui suivent l'obtention de ma note.

Question 2

Terminé

Non noté

Si nécessaire, inscrivez vos suppositions ou commentaires ici, en précisant pour chaque supposition le numéro de la question concernée.

Question 7: « Soit le programme suivant. Complétez chaque case de manière à ce que l'exécution du programme corresponde à l'affichage.

```
vec = [10, 20, 30, 40] [...]
```

On dirait qu'on souhaite faire une sorte de tri? Mais, puisque `j` est toujours plus grand que `i` (`j` est fait à partir de `range(i + 1, ...)`), les valeurs de `vec` sont croissantes. Alors `vec[j] <= vec[i]` va toujours être faux (avec un pas positif). Ainsi, le résultat attendu sera toujours le même, peu importe les réponses données (évidemment, tant qu'il n'y ait pas d'erreurs de syntaxe). J'ai décidé de mettre des valeurs plausibles qui

ressemblent à un algorithme de triage, et pour la dernière réponse j'ai mis une valeur similaire à l'affectation d'en-dessous.

Question 3

Terminé

Note de 0,50
sur 0,50

Quel sera le type du résultat de la division (opérateur `/`) entre deux nombres entiers (`int`)?

- ☐ a. Entier (`int`)
- ☐ b. Chaîne de caractère (`str`)
- ☒ c. Réel (`float`)

Question 4

Terminé

Note de 0,00
sur 0,50

Une tranche (ex: `ma_liste[1:6:2]`) permet de sélectionner n'importe quel sous-ensemble d'une liste ou chaîne de caractère.

Veillez choisir une réponse.

- ☒ Vrai
- ☐ Faux

Question 5

Terminé

Note de 0,50
sur 0,50

Les clés d'un dictionnaire ne peuvent être que des types hachables et immuables. Les valeurs d'un dictionnaires peuvent être de n'importe quel type.

Veillez choisir une réponse.

- ☒ Vrai
- ☐ Faux

Question 6

Terminé

Note de 0,50
sur 1,00

Soit le programme suivant. Complétez chaque case de manière à ce que l'exécution du programme corresponde à l'affichage. Assurez-vous que la variable `fifo` soit bien utilisée comme une file (*queue*) avec seulement les opérations qui s'appliquent sur cette structure de donnée.

```
fifo = deque()
l=[10,20,30,40]
for i in [10, 20, 30, 40]:
    fifo. extend([30, 40])

x = fifo. pop()
x = fifo. popleft()
y = fifo. popleft()

print("x vaut ", x, " et y vaut ", y)
```

Affichage:

x vaut 30 et y vaut 40

Question 7

Terminé

Note de 1,00

Soit le programme suivant. Complétez chaque case de manière à ce que l'exécution du programme corresponde à l'affichage.

sur 1,00

```
vec = [10, 20, 30, 40]

for i in range(0, 4):
    for j in range(i + 1, 4):
        if vec[j] <= vec[i]:
            vec[j] = 0
            vec[i] = 0

print(vec[0], "-", vec[1], "-", vec[3])
```

Affichage:

10 - 20 - 40

Question 8

Terminé

Note de 1,00
sur 1,00

Soit le programme suivant. Complétez chaque case de manière à ce que l'exécution du programme corresponde à l'affichage.

```
D = {1: 30, 2: 45, 3: 21, 4: 11, 5: 10}
D.pop( 3 )
for element in reversed (D):
    if D[element] % 2 != 0:
        print(f"{element}: {D[element]}")
```

Affichage :

4: 11
2: 45

Question 9

Terminé

Note de 2,50
sur 2,50

Soit la fonction suivante :

```
def truc(a, b=0, c=1):
    return a + b, b / c
```

Cochez tous les appels de fonctions valides, c'est-à-dire ceux qui vont pas mener à une erreur.

Veuillez choisir au moins une réponse.

- ☒ resultat = truc(1, 2, 3)
- ☒ abc = truc(1, 2)
- ☐ truc({a: 1, c: 4, b: 5})
- ☐ abc = truc()
- ☐ crut(1, 2, 3)
- ☐ truc(a=1, 4, 5)
- ☒ abc = truc(10)
- ☒ a, b = truc(a=3, b=5, c=6)
- ☒ c, d = truc(*(1, 2, 3))

☒ res = truc(c=5, a=4, b=3)

Question 10

Correct

Note de 5,00
sur 5,00

Soit l'approximation de $\pi/2$ par la transformation de Newton/Euler :

$$\frac{\pi}{2} = \sum_{k=0}^{\infty} \frac{2^k k!^2}{(2k+1)!} = 1 + 0.33333 + 0.13333 + 0.05714 + \dots$$

Écrivez la fonction `newton_euler` qui prend en paramètre *epsilon* (un réel) et retourne une approximation de la valeur de $\pi/2$ en utilisant la formule ci-haut. On arrête la somme au terme qui est inférieur à *epsilon*.

Par exemple, pour un *epsilon* de 1, on arrête au deuxième terme 0.3333 qui est < 1, on obtient donc le résultat 1.3333. Si on applique un epsilon de 0.2, on arrête au troisième terme, car 0.1333 < 0.2, donc résultat de 1.46667. Dans les tests fournis les nombres sont arrondis.

N'oubliez pas que vous avez accès à `math.factorial`.

Par exemple:

Test	Résultat
<code>print(f"{newton_euler(1):.5f}")</code>	1.33333
<code>print(f"{newton_euler(0.2):.5f}")</code>	1.46667

Réponse : (régime de pénalités : 0 %)

Réinitialiser la réponse

```
1 def newton_euler(epsilon: int) -> float:
2     result = 0
3     n = 0
4     new_value = math.inf
5
6     while True:
7         if new_value < epsilon:
8             break
9
10        new_value = (2**n) * (math.factorial(n) ** 2) / math.factorial(2 * n)
11        result += new_value
12        n += 1
13
14    return result
```

	Test	Résultat attendu	Résultat obtenu	
✓	<code>print(f"{newton_euler(1):.5f}")</code>	1.33333	1.33333	✓
✓	<code>print(f"{newton_euler(0.2):.5f}")</code>	1.46667	1.46667	✓
✓	<code>print(f"{newton_euler(0.01):.5f}")</code>	1.56608	1.56608	✓

Tous les tests ont été réussis ! ✓

Correct

Note pour cet envoi : 5,00/5,00.

Question 11

Correct

Note de 4,00
sur 4,00

Considérez la fonction suivante, dont l'objectif est de modifier une liste en supprimant les valeurs supérieures à un seuil donné en paramètre.

```
1. def modifier_liste(liste, seuil):
2.     """
3.     Cette fonction supprime les elements strictement superieurs a un seuil d'une liste de nombres
4.     :param liste: la liste de nombres à modifier (entier ou flottant)
5.     :param seuil: nombre (entier ou flottant)
6.     """
7.     for i in range(len(liste)):
8.         if liste[i] > seuil:
9.             del liste[i]
```

Cette fonction n'implémente pas le comportement désiré à cause d'un problème de logique. Votre objectif est de proposer une solution alternative permettant de supprimer tous les éléments de la liste strictement supérieurs au seuil donné en paramètre, sans causer d'erreur.

Par exemple:

Test	Résultat
ma_liste = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9] modifier_liste(ma_liste, seuil=6) print(ma_liste)	[0, 1, 2, 3, 4, 5, 6]
ma_liste = [0, 1, 3, 7, 5, 8, 3, 4, 1, 9, 3] modifier_liste(ma_liste, seuil=5) print(ma_liste)	[0, 1, 3, 5, 3, 4, 1, 3]

Réponse : (régime de pénalités : 0 %)

Réinitialiser la réponse

```
1 def modifier_liste(liste: list, seuil: int):
2     """
3     Cette fonction supprime les elements strictement superieurs a un seuil d'un
4     :param liste: la liste de nombres à modifier (entier ou flottant)
5     :param seuil: nombre (entier ou flottant)
6     """
7     # on veut changer directement la liste sans en créer une autre
8     liste[:] = [nombre for nombre in liste if not nombre > seuil]
```

	Test	Résultat attendu	Résultat obtenu	
✓	ma_liste = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9] modifier_liste(ma_liste, seuil=6)	[0, 1, 2, 3, 4, 5, 6]	[0, 1, 2, 3, 4, 5, 6]	✓

	<code>print(ma_liste)</code>			
✓	<code>ma_liste = [0, 1, 3, 7, 5, 8, 3, 4, 1, 9, 3]</code> <code>modifier_liste(ma_liste, seuil=5)</code> <code>print(ma_liste)</code>	<code>[0, 1, 3, 5, 3, 4, 1, 3]</code>	<code>[0, 1, 3, 5, 3, 4, 1, 3]</code>	✓
✓	<code>ma_liste = [0]</code> <code>modifier_liste(ma_liste, seuil=5)</code> <code>print(ma_liste)</code>	<code>[0]</code>	<code>[0]</code>	✓
✓	<code>ma_liste = [10]</code> <code>modifier_liste(ma_liste, seuil=5)</code> <code>print(ma_liste)</code>	<code>[]</code>	<code>[]</code>	✓
✓	<code>ma_liste = [-1, 2, -3, -5, -6, 4, 0]</code> <code>modifier_liste(ma_liste, seuil=-3)</code> <code>print(ma_liste)</code>	<code>[-3, -5, -6]</code>	<code>[-3, -5, -6]</code>	✓

Tous les tests ont été réussis ! ✓

Correct

Note pour cet envoi : 4,00/4,00.

Question 12

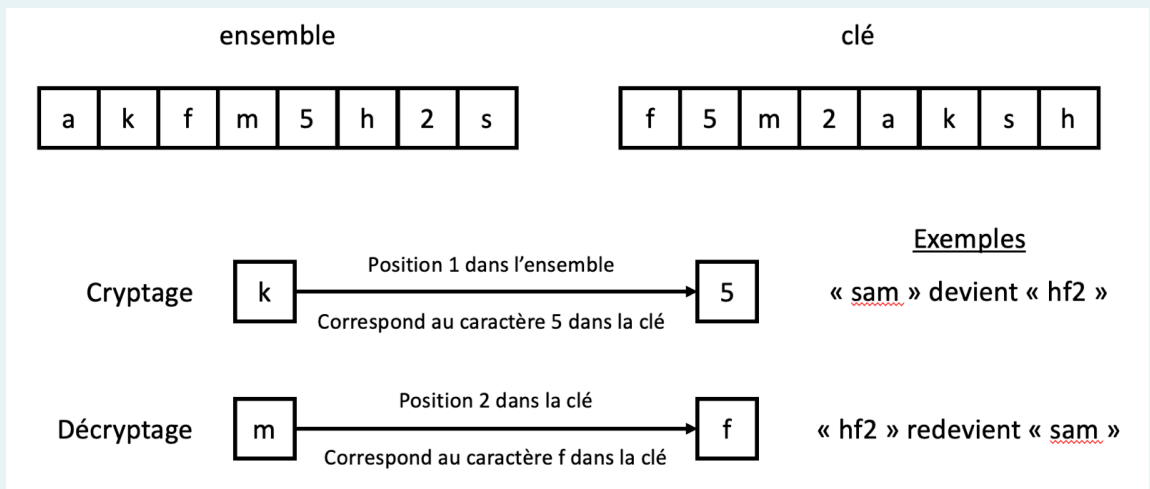
Correct

Note de 4,00
sur 4,00

Vous venez d'obtenir votre premier emploi dans une célèbre entreprise de cybersécurité. Bravo!

Votre premier mandat consiste en l'implémentation d'un algorithme de décryptage par clé à usage unique. Cet algorithme de cryptage et de décryptage se base sur l'utilisation de deux éléments : un ensemble de caractère et une clé générée aléatoirement à partir de cet ensemble. L'algorithme de décryptage suit la procédure suivante :

Soit un texte secret qui a été encrypté. Pour chaque caractère du texte secret, on recherche la position de ce caractère dans la clé. Par la suite, on utilise la position trouvée pour récupérer le caractère correspondant dans l'ensemble de caractère fourni. L'image ci-dessous illustre le fonctionnement de l'algorithme d'encryptage et de décryptage.



Afin de vous aider dans votre tâche, votre collègue vous a fourni la fonction permettant d'encoder une chaîne de caractère avec une clé. Vous devez implémenter la fonction `decode` afin de retrouver le message initial qui a été encodé. Cette fonction prend en paramètre le texte secret, la clé et l'ensemble (toutes des string). Le paramètre `ensemble` a comme valeur par défaut `string.printable`.

Notes :

- L'ensemble par défaut est l'ensemble des caractères imprimable (`string.printable`)
- Les modules `random` et `string` ont déjà été importés
- La clé est générée aléatoirement à partir de l'ensemble pour chaque validation, en utilisant l'expression suivante : `cle = "".join(random.sample(string.printable, len(string.printable)))`

Par exemple:

Test	Résultat
<pre>ensemble = "akfm5h2s" cle = "f5m2aksh" texte_encode = encode("sam", cle, ensemble) print(texte_encode) print(decode(texte_encode, cle, ensemble))</pre>	hf2 sam
<pre>ensemble = "0123456789" cle = "9876543210" texte_encode = encode("1337", cle, ensemble) print(texte_encode) print(decode(texte_encode, cle, ensemble))</pre>	8662 1337
<pre>cle = "".join(random.sample(string.printable, len(string.printable))) texte_secret = encode("Je vais reussir l'examen!", cle) print(decode(texte_secret, cle))</pre>	Je vais reussir l'examen!

Réponse : (régime de pénalités : 0 %)

Réinitialiser la réponse

```
1 def encode(texte, cle, ensemble=string.printable):
2     # Ceci est fait pour vous, rien à modifier
3     return "".join([cle[ensemble.find(caractere)] for caractere in texte if c
4
5 def decode(texte_secret: str, cle: str, ensemble: str = string.printable) ->
6     return "".join(
7
8         [
9             ensemble[cle.find(caractere)]
10            for caractere in texte_secret
11            if caractere in cle
12        ]
13    )
```

	Test	Résultat attendu	Résultat obtenu	
✓	<pre>ensemble = "akfm5h2s" cle = "f5m2aksh" texte_encode = encode("sam", cle, ensemble) print(texte_encode) print(decode(texte_encode, cle, ensemble))</pre>	hf2 sam	hf2 sam	✓
✓	<pre>ensemble = "0123456789" cle = "9876543210" texte_encode = encode("1337", cle, ensemble) print(texte_encode) print(decode(texte_encode, cle, ensemble))</pre>	8662 1337	8662 1337	✓
✓	<pre>cle = "".join(random.sample(string.printable, len(string.printable))) texte_secret = encode("Je vais reussir l'examen!", cle)</pre>	Je vais reussir l'examen!	Je vais reussir l'examen!	✓

```
print(decode(texte_secret, cle))
```

Tous les tests ont été réussis ! ✓

Correct

Note pour cet envoi : 4,00/4,00.

◀ [Contrôle périodique - Été 2021](#)
(pour révision)

Aller à...

[Simulation Examen \(Groupe5\)](#) ▶