Commencé le	Monday 25 March 2024, 11:02
État	Terminé
Terminé le	Monday 25 March 2024, 11:27
Temps mis	24 min 35 s
Note	**20,00** sur 20,00 (**100**%)

Question **1**

Correct

Note de 2,00 sur 2,00

Retourner une liste avec toutes les valeurs du bord d'un tableau 2D. L'ordre des valeurs n'est pas important.

Return a list with all the edge values of a 2D array. The order of the values is not important.

Exemple : si A = np.array([[1,2,3],[4,5,6],[7,7,7]]) alors on retourne [1,2,3,4,6,7,7,7]

En 3 lignes max

Réponse : (régime de pénalités : 0 %)

Réinitialiser la réponse

```
def couronne(A):
    return list(A[[0, -1], :].flatten()) + list(A[1:-1, [0, -1]].flatten())
```

Solution de l'auteur de la question (Python3):

```
def couronne(A):
    return list(A[[0,-1],:].flatten()) +   list(A[1:-1, [0,-1]].flatten())
```

Correct

Note pour cet envoi : 2,00/2,00.

Question **2**

Correct

Note de 2,00 sur 2,00

Ecrire une fonction qui échange les colonnes i et j d'un tableau A.

Write a function which swap columns i and j of a array A.

Nombre de lignes maximum = 3

Réponse : (régime de pénalités : 0 %)

Réinitialiser la réponse

```
def swap_col(A, i, j):
    A[:,[i,j]] = A[:,[j,i]]
    return A
```

Solution de l'auteur de la question (Python3):

```
def swap_col(A, i, j):
    A[:,[i,j]] = A[:,[j,i]]
    return A
```

Correct

Note pour cet envoi : 2,00/2,00.

Question **3**

Correct

Note de 2,00 sur 2,00

Écrire une fonction qui indique quelle est la valeur d'un vecteur la plus proche d'une valeur `a` donnée en argument.

Write a function that returns the closest value of a vector from a given value `a`.

[In] : closest(np.arange(5), 2.1)

[Out] : 2

2 lignes max

Réponse : (régime de pénalités : 0 %)

Réinitialiser la réponse

```
def closest(v, a):
    return v[np.argmin(np.abs(v - a))]
```

Solution de l'auteur de la question (Python3):

```
def closest(v, a):
    return v[np.argmin(np.abs(v - a))]
```

Correct

Note pour cet envoi : 2,00/2,00.

Question **4**

Correct

Note de 2,00 sur 2,00

Trouver la plus petite valeur impaire d'une matrice. S'il n'y a pas de valeur impaire, retourner None.

Find the smallest odd value of a matrix. If there is no odd value, return None.

Nombre de lignes maximum = 2

Réponse : (régime de pénalités : 0 %)

Réinitialiser la réponse

```
def small_odd(A):
    A[A%2!=0].min() if np.any(A%2!=0) else None
```

Solution de l'auteur de la question (Python3):

```
def small_odd(A):
    return A[A % 2 ==1].min() if np.any(A % 2 ==1) else None
```

Correct

Note pour cet envoi : 2,00/2,00.

Question **5**

Correct

Note de 2,00 sur 2,00

Soit un vecteur v, on désire créer la matrice A qui a

- v en 1ere colonne,
- v décalé d'un cran en 2e colonne (A[0,1] = v[1] et A[-1,1] = v[0]),
- v décalé de 2 crans en 3e colonne (A[0,2] = v[2] et A[-1,2] = v[1]),

etc

`for` autorisé.

Create a array A from a known vect v such that

- 1rst column of A is v
- 2nd column of A is v shifted by 1
- 3rd column of A is v shifted by 3
- etc

`for`is allowed for this question.

Nombre max de lignes = 2

Réponse : (régime de pénalités : 0 %)

Réinitialiser la réponse

```
def matspe(v):
    return np.stack([np.roll(v, -i) for i in range (len(v))], axis=1)
```

Solution de l'auteur de la question (Python3):

```
def matspe(v):
    return np.stack([np.roll(v,-i) for i in range(len(v))], axis=1)
```

Correct

Note pour cet envoi : 2,00/2,00.

Question **6**

Correct

Note de 2,00 sur 2,00

Transformer un tableau pour que sa moyenne soit 0 et son écart type égal à 1.

Center values of an array on 0 with a standart deviation of 1.

Réponse en 2 lignes de moins de 35 caractères

Réponse : (régime de pénalités : 0 %)

Réinitialiser la réponse

```
def normalize(A):
    return (A - A.mean()) / A.std()
```

Solution de l'auteur de la question (Python3):

```
def normalize(A):
    return (A - A.mean()) / A.std()
```

Correct

Note pour cet envoi : 2,00/2,00.

Question **7**

Correct

Note de 2,00 sur 2,00

Soit C = A B le produit matriciel des matrices A et B stockées comme deux tableaux Numpy. Écrire la fonction optimale qui retourne la valeur C[i,j]

Les C = AB the matrix product of matrices A and B stored in Numpy arrays. Write the optimal function which returns C[i,j].

En 2 lignes de 30 caractères max

Réponse : (régime de pénalités : 0 %)

Réinitialiser la réponse

```
def C(i, j, A, B):
    return A[i,:] @ B[:,j]
```

	Test	Résultat attendu	Résultat obtenu	

✔	`A = np.random.randint(-10, 10, size=(6,6))`
	`B = np.random.randint(-10, 10, size=(6,6))`
	`print(C(4,2,A,B) == A[4,:] @ B[:,2])`
		`True`	`True`	✔

Tous les tests ont été réussis ! ✔

Solution de l'auteur de la question (Python3):

```
def C(i, j, A, B):
    return A[i,:] @ B[:,j]
```

Correct

Note pour cet envoi : 2,00/2,00.

Question **8**

Correct

Note de 2,00 sur 2,00

Écrire une fonction qui échange la valeur maximale d'un tableau avec sa valeur minimale.

Write a function that swap the min and the max of an array.

Nombre de lignes max = 3

Réponse : (régime de pénalités : 0 %)

Réinitialiser la réponse

```
def maxmin(A):
    A.flat[[A.argmax(), A.argmin()]] = A.min(), A.max()
    return A
```

Solution de l'auteur de la question (Python3):

```
def maxmin(A):
    A.flat[[A.argmax(),A.argmin()]] = A.min(), A.max()
    return A
```

Correct

Note pour cet envoi : 2,00/2,00.

Question **9**

Correct

Note de 2,00 sur 2,00

Ranger les colonnes d'un tableau dans l'ordre croissant de leur moyenne.

Sort columns of an array by their mean in ascending order.

max 2 lignes

Réponse : (régime de pénalités : 0 %)

Réinitialiser la réponse

```
def sort_col(A):
    return A[:, np.argsort(np.mean(A, axis=0))]
```

	Test	Résultat attendu	Résultat obtenu	

✔	`A = np.random.randint(0,20,size=(8,7))`
	`B = A.copy()`
	`res = sort_col(A)`
	`B = B[:, A.sum(axis=0).argsort()]`
	`print(np.abs(res - B).sum() < 1E-6)`
		`True`	`True`	✔

Tous les tests ont été réussis ! ✔

Solution de l'auteur de la question (Python3):

```
def sort_col(A):
    return A[:, A.sum(axis=0).argsort()]
```

Correct

Note pour cet envoi : 2,00/2,00.

Question **10**

Correct

Note de 2,00 sur 2,00

Soit une liste de points en 2D qu'on peut imaginer comme les coordonnées (x,y) de villes. On veut un tableau D qui donne la distance à vol d'oiseau entre 2 villes. Donc a D[i,j] = D[j,i] = distance entre les villes (points) i et j.

Les n points sont donnés dans une matrice (n,2). Écrire la fonction qui prend ces points et retourne la matrice D des distances.

Let's have a list of 2D points which can be thought of as the (x,y) coordinates of cities. We want an array D that gives the bird's-eye view distance between 2 cities. Thus, at D[i,j] = D[j,i] = distance between cities (points) i and j.

The n points are given in a matrix (n,2). Write the function that takes these points and returns the matrix D of distances.

Nombre de ligne max = 4

Réponse : (régime de pénalités : 0 %)

Réinitialiser la réponse

```
def distpoints(P):
    x = np.array([P[:,0]])
    y = np.array([P[:,1]])
    return np.sqrt(np.square(x - x.T) + np.square(y - y.T))
```

Solution de l'auteur de la question (Python3):

```
def distpoints(P):
    x = np.array([P[:,0]])
    y = np.array([P[:,1]])
    return np.sqrt(np.square(x - x.T) + np.square(y - y.T))
```

Correct

Note pour cet envoi : 2,00/2,00.

◄ Avertissement

Aller à…

Examen de CAMA ►