|  |  |  |
| --- | --- | --- |
| Spécialité NSI Première | **DEVOIR SURVEILLE DE** | Vendredi 24 novembre 2023 |
| Lycée d’Avesnières | **NSI** | Durée : 2 h |
| Année scolaire 2023-2024 | **N° 2** | Calculatrice interdite |

**NOM : ..........................................................**

**Prénom : ......................................................**

**Rendre l'énoncé avec la copie.**

**Exercice 1** ( 2 points)

Si a vaut False et b vaut True, déterminer, en justifiant, la valeur des expressions booléennes suivantes :

1. a or b and not a.
2. a and (b or not a).

**Exercice 2** ( 2,5 points)

Jeanne a trouvé une des lois de De Morgan qui permet de simplifier une expression booléenne :

not(a and b) = not a or not b

Elle se pose la question : « Cette loi est-elle correcte ? »

Vous remplirez sur cet énoncé les tables de vérité suivantes puis vous répondrez à la question sur votre copie.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| a | b | a and b | not(a and b) |  | a | b | not a | not b | not a or not b |
| False | False |  |  |  | False | False |  |  |  |
| False | True |  |  |  | False | True |  |  |  |
| True | False |  |  |  | True | False |  |  |  |
| True | True |  |  |  | True | True |  |  |  |

**Exercice 3** ( 5,5 points)

**Partie A**

Paul a écrit cette page HTML :

Une image contenant texte

Description générée automatiquementUne image contenant texte

Description générée automatiquement

1. Lorsqu'il ouvre cette page sur son navigateur, il voit l'affichage :

Que peut-il remarquer ?

1. Paul a installé un éditeur hexadécimal qui lui permet de voir le contenu binaire de son fichier html sous la forme d'une succession d'octets. L'octet 20 est le code du caractère "espace entre les mots".

Une image contenant table

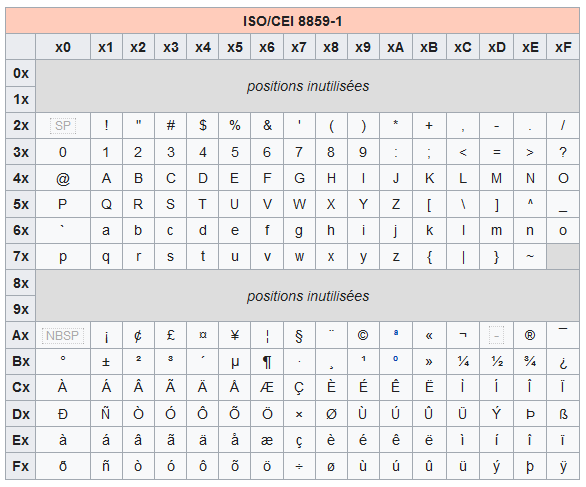
Description générée automatiquement

* 1. Écrivez les huit bits de l'octet qui sert à coder le caractère "espace".
  2. L'éditeur hexadécimal affiche le codage du fichier par des lignes contenant chacune seize octets. Il y a correspondance entre les lignes de codage et les lignes de texte affiché à droite.

Écrivez la ligne d'octets, en hexadécimal, qui correspond au texte "en € des gains"

* 1. Comment est codé le caractère '€' dans le fichier ?

1. Voici la table d'encodage de caractères ISO 8859-1 :



En utilisant la table, justifiez l'affichage "â ¬" visible dans le navigateur.

1. Le fichier HTML de Paul a-t-il été encodé selon l'encodage des caractères ISO8859-1 ? Justifiez.
2. Selon quel encodage de caractères Paul a-t-il enregistré le fichier texte de sa page html ?
3. Que conseillez-vous à Paul pour corriger ce problème ?

**Partie B**

1. En utilisant la partie A, écrivez les 3 octets qui codent le symbole de l'euro.

Écrivez les 24 bits qui correspondent à ces trois octets.

1. Parmi ces 24 bits, certains sont "fixes". Les autres, sont les bits 'b' qui servent à écrire le point de code [[1]](#footnote-1)Unicode du caractère. La figure ci-contre donne la position des bits 'b'.

1110bbbb 10bbbbbb 10bbbbbb

* 1. Écrivez la suite des bits servant à former le point de code Unicode du symbole €.

En rouge les bits fixes, en vert ceux de l'index Unicode.

* 1. Déduisez-en le point de code dans le standard Unicode du symbole € en écriture décimale.

**Exercice 4** ( 4,5 points)

Un carré d'ordre est un tableau carré contenant entiers strictement positifs. On dit qu'un carré d'ordre est magique si :

il contient tous les nombres entiers depuis jusqu'à inclus. De plus, il faut que :

* La somme des nombres sur chaque ligne soit la même.
* La somme des nombres sur chaque colonne soit la même.
* La somme des nombres sur chaque diagonale soit la même.
* Les trois sommes précédentes (lignes, colonnes, diagonales) soient les mêmes.

***Exemple 1 :*** le carré contient tous les entiers de à inclus et toutes les sommes valent 15.

|  |  |  |  |
| --- | --- | --- | --- |
| 15 | 15 | 15 | 15 |
| 15 |  |  |  |
| 15 |  |  |  |
| 15 |  |  |  |

On modélise le carré par la variable carre3 = [[2, 7, 6], [9, 5, 1], [4, 3, 8]]

***Exemple 2 :*** le carré magique suivant est un carré d'ordre

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Pour le modéliser, on crée la variable

carre4 = [[4, 5, 11, 14], [15, 10, 8, 1], [6, 3, 13, 12], [9, 16, 2, 7]]

1. a. Quel est le type de la variable carre4 ?

b. Quelle est la valeur de len(carre4) ?

c. Quelle est la valeur de carre3[1] ?

d. Quelle est la valeur de carre3[0][2] ?

e. Quelle expression permet de récupérer la valeur qui est dans la variable carre4 ?

1. On donne la fonction ci-contre :

def somme\_ligne(carre, i):

    """

    Paramètres :

    ------------

    carre : de type liste de listes.

            Elle représente un tab-

            leau carré de nombres.

    i : de type entier.

    """

    somme = 0

    for nombre in carre[i]:

        somme = somme + nombre

    return somme

* 1. Quelle valeur renvoie somme\_ligne(carre4, 2) ?
  2. A quoi sert la fonction somme\_ligne ?

1. Proposer, en langage Python, une fonction verifie\_lignes, prenant en paramètres

* une liste de listes représentant un tableau carré de nombres entiers
* un entier égal au côté du tableau carré

et qui renvoie un booléen indiquant l'égalité des sommes sur toutes les lignes ou non. Par exemple :

>>> verifie\_lignes(carre3, 3)

True

>>> verifie\_lignes(carre4, 4)

True

**Exercice 5** ( 5,5 points)

1. Donnez sept types de variables déjà rencontrés en précisant s’il s’agit d’un type de base ou d’un type construit.
2. Quelle instruction permet d’associer une valeur à une variable et quel est le symbole dédié en Python ?
3. Donnez un exemple d’association de valeur pour chacun des sept types de variables.
4. A partir des scripts Python suivants,

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| def trouver\_lettre(phrase, lettre):      index\_resultat = 0      for i in range(len(phrase)):          if phrase[i] == lettre:              index\_resultat = i      return index\_resultat |  | def rebond(g, h):      d = False      if g > h + 10:          d = True      return d |  | def f(x):      return(x, x\*\*2)  a = f(4) |
|  |
|  |
|  |  |  |  |  |
| t = [{'id': 1, 'age': 23}, {'id': 2, 'age': 27}, {'id': 4, 'age': 41}]  r = [c for c in t if c['age'] > 30] | | | | |

remplissez le tableau suivant **en cochant par une croix** le ou les types possibles pour chacune des variables utilisées.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | index\_resultat | i | lettre | phrase | t | c | r | d | g | h | a | x |
| int |  |  |  |  |  |  |  |  |  |  |  |  |
| str |  |  |  |  |  |  |  |  |  |  |  |  |
| bool |  |  |  |  |  |  |  |  |  |  |  |  |
| float |  |  |  |  |  |  |  |  |  |  |  |  |
| list |  |  |  |  |  |  |  |  |  |  |  |  |
| dict |  |  |  |  |  |  |  |  |  |  |  |  |
| tuple |  |  |  |  |  |  |  |  |  |  |  |  |

1. **Point de code :** c'est le numéro que porte un caractère dans le standard Unicode. [↑](#footnote-ref-1)