

Vers une classification périodique

Objectif : Réaliser un programme affichant la classification périodique simplifiée des 18 premiers éléments chimiques

Une classification périodique simplifiée des 18 premiers éléments chimiques est présentée ci-dessous :

Numéro atomique

Nom de l'élément chimique

Symbole de l'élément chimique

Structure électronique de l'atome

| | | | | | | | |
|---|--|--|---|---|--|---|--|
| 1 H Hydrogène K ¹ | | | | | | | 2 He Hélium K ² |
| 3 Li Lithium K ² L ¹ | 4 Be Béryllium K ² L ² | 5 B Bore K ² L ³ | 6 C Carbone K ² L ⁴ | 7 N Azote K ² L ⁵ | 8 O Oxygène K ² L ⁶ | 9 F Fluor K ² L ⁷ | 10 Ne Néon K ² L ⁸ |
| 11 Na Sodium K ² L ⁸ M ¹ | 12 Mg Magnésium K ² L ⁸ M ² | 13 Al Aluminium K ² L ⁸ M ³ | 14 Si Silicium K ² L ⁸ M ⁴ | 15 P Phosphore K ² L ⁸ M ⁵ | 16 S Soufre K ² L ⁸ M ⁶ | 17 Cl Chlore K ² L ⁸ M ⁷ | 18 Ar Argon K ² L ⁸ M ⁸ |

Votre travail : Réaliser un programme permettant un affichage similaire de la classification périodique simplifiée des 18 premiers éléments chimiques

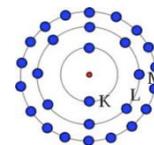
Quelques rappels de chimie :

Numéro atomique d'un élément chimique: Le numéro atomique noté Z correspond aux nombres de protons que contient le noyau des différentes entités appartenant à cet élément.

Nombre d'électrons contenus dans un atome: Un atome contient autant d'électrons que de protons donc Z électrons.

Structure électronique d'un atome :

Dans le modèle de l'atome de Bohr, les électrons, situés autour du noyau, gravitent sur des orbites particulières appelées couches électroniques. La répartition des électrons dans les différentes couches est appelée structure électronique de l'atome.



En classe de seconde seules les structures électroniques des 18 premiers éléments chimiques sont étudiées. Ainsi, les atomes considérés posséderont entre 1 et 18 électrons répartis dans les 3 premières couches K, L et M.

Règles de remplissage valables pour les 18 premiers éléments chimiques (numéros atomiques Z allant de 1 à 18) : Les électrons se répartissent d'abord sur la couche K, puis s'il reste des électrons, ils remplissent la couche L, puis la couche M.

- La couche K : contient jusqu'à 2 électrons
- La couche L : contient jusqu'à 8 électrons
- La couche M : contient jusqu'à 18 électrons

Exemple : La structure de l'atome de lithium (Symbole Li, Numéro atomique Z= 3) est K² L¹

Solution guidée :

En fonction des difficultés rencontrées, les élèves peuvent suivre la progression suivante :

Partie 1 : Vers l'écriture d'une fonction déterminant la structure électronique d'un atome

Programme 1: Ecrire un programme qui détermine et affiche la structure électronique d'un atome à partir de son numéro atomique Z ($Z \leq 18$).

L'utilisateur entre le numéro atomique de l'atome considéré

Le programme affiche la structure électronique sous la forme K2 L8 M5

Correction Programme 1:

```
1 # Programme affichant la structure électronique d'un atome
2 Z= int(input("Entrer le numéro atomique de l'atome"))
3 if (Z<=2):
4     print("K",Z)
5 if (2<Z<=10):
6     print("K2 L",Z-2)
7 if (10<Z<=18):
8     print("K2 L8 M",Z-10)
```

Programme 2: Modifier le programme précédent afin d'écrire une fonction qui renvoie la structure électronique d'un atome à partir d'un numéro atomique passé en paramètre.

- La fonction renvoie une chaîne de caractère contenant la structure électronique sous la forme K2 L8 M5

INFO : Concaténation de chaîne de caractères

Le programme suivant

```
1 age=21
2 message= "J'ai " + str(age) + " ans"
3 print (message)
```

affiche

```
>>>
J'ai 21 ans
>>>
```

Correction Programme 2:

```
3 def structure(Z):
4     if (Z<=2):
5         rep = "K"+str(Z)
6     if (2<Z<=10):
7         rep="K2 L"+ str(Z-2)
8     if (10<Z<=18):
9         rep="K2 L8 M"+str(Z-10)
10    return (rep)
```

Partie 2 : Vers la création de la liste des 18 premiers éléments chimiques

Programme 3 : Ecrire un programme dans lequel deux listes sont créées : une contenant la liste des noms des 18 premiers éléments chimiques puis une deuxième contenant leur symbole.

Correction Programme 3:

```
13 Symbole= ["H", "He", "Li", "Be", "B", "C", "N", "O", "F", "Ne", "Na", "Mg", "Al",  
14           "Si", "P", "S", "Cl", "Ar"]  
15  
16 nom= ["Hydrogène", "Helium", "Lithium", "Beryllium", "Bore", "Carbone",  
17       "Azote", "Oxygène", "Fluor", "Neon", "Sodium", "Magnésium", "Aluminium",  
18       "Silicium", "Phosphore", "Soufre", "Chlore", "Argon"]  
--
```

Partie 3 : Vers le dessin du tableau → Interface graphique avec Tkinter

Le module Tkinter ("Tk interface") de Python permet de créer des interfaces graphiques

De nombreux composants graphiques (ou widgets) sont disponibles : fenêtre (classe Tk), bouton (classe Button), case à cocher (classe Checkbutton), étiquette (classe Label), zone de texte simple (classe Entry), menu (classe Menu), zone graphique (classe Canvas), cadre (classe Frame)...

Nous allons nous intéresser plus particulièrement à la classe Canvas

Un canevas est une zone rectangulaire destinée à contenir des dessins ou d'autres figures complexes. Vous pouvez y placer des graphiques, du texte, des composants graphiques (*widgets*) ou des cadres (*frames*).

```
Canvas.create_rectangle(x0, y0, x1, y1, option, ...)
```

```
Canvas.create_text(x, y, option, ...)
```

Programme 4: Ecrire un programme permettant la construction d'un rectangle.

Programme 5: Ecrire un programme permettant la construction d'un tableau contenant 8 colonnes et 3 lignes.

Programme 6 : Compléter le programme précédent de sorte que le mot « symbole » soit écrit dans chacune des cases du tableau.

Partie 4 : Vers l'objectif final

En utilisant l'ensemble des programmes vus précédemment, écrire le programme permettant de répondre au problème initial.

| | | | | | | | |
|--------------------------|-----------------------------|-----------------------------|----------------------------|----------------------------|-------------------------|--------------------------|-------------------------|
| H Hydrogène K1 | | | | | | | He Helium K2 |
| Li Lithium K2 L1 | Be Beryllium K2 L2 | B Bore K2 L3 | C Carbone K2 L4 | N Azote K2 L5 | O Oxygène K2 L6 | F Fluor K2 L7 | Ne Neon K2 L8 |
| Na Sodium K2 L8 M1 | Mg Magnésium K2 L8 M2 | Al Aluminium K2 L8 M3 | Si Silicium K2 L8 M4 | P Phosphore K2 L8 M5 | S Soufre K2 L8 M6 | Cl Chlore K2 L8 M7 | Ar Argon K2 L8 M8 |

Partie 5 : Pour aller plus loin

- Ecrire une fonction qui renvoie la formule de l'ion monoatomique stable qu'un élément chimique va avoir tendance à former. L'utilisateur entrera le numéro atomique (et le symbole ? à moins que cela soit un prolongement) de l'élément chimique. Pour $Z=3$, le programme affiche Li 1+
- Indiquer dans le tableau ci-dessus l'ion monoatomique stable que chaque atome va préférentiellement former.

Banque d'exercices autour de la classification périodique

Scenario 2 :

niveau 1 : Créer une fonction, qui pour n (Z), le numéro atomique entré, renvoie le symbole chimique et le nom de l'atome.

```
31 def atome(n):
32     mendeleiv=["H","He","Li","Be","B","C","N","O","F","Ne","Na","Mg","Al"]
33     nom=["Hydrogène","Hélium","Lithium","Béryllium","Bore","Carbone","Azote"]
34     return(mendeleiv[n-1],nom[n-1])
```

niveau 2 : Créer une fonction, qui pour n (Z), le numéro atomique entré, renvoie le symbole chimique et le nom de l'atome et précise si c'est la cas qu'il 'agit d'un gaz rare, d'un métal alcalin pu d'un halogène.

```
17
18 def atome(n):
19     mendeleiv=["H","He","Li","Be","B","C","N","O","F","Ne","Na","Mg","Al"]
20     nom=["Hydrogène","Hélium","Lithium","Béryllium","Bore","Carbone","Azote"]
21     famille=["gaz noble","métal alcalin","halogène"]
22     if n == 2 or n==10 or n==18 :
23         return(mendeleiv[n-1],nom[n-1],famille[0])
24     elif n== 3 or n == 11 :
25         return(mendeleiv[n-1],nom[n-1],famille[1])
26     elif n == 9 or n ==17 :
27         return(mendeleiv[n-1],nom[n-1],famille[2])
28     else :
29         return(mendeleiv[n-1],nom[n-1])
```

niveau 3 :

```
34 def choix_signe(k,n) :
35     signe=["+","-"]
36     if k<n :
37         signe=signe[1]
38     else :
39         signe=signe[0]
40     return(signe)
41
42
43
44 def transfor(n):
45     stable=["He","C","Ne","Si","Ar"]
46     if n < 6 and n!=2 :
47         k=2
48         return(stable[0],abs(n-k),choix_signe(k,n))
49     elif n >6 and n< 14 and n!=10 :
50         k=10
51         return(stable[2],abs(n-k),choix_signe(k,n))
52     elif n > 14 and n!=18 :
53         k=18
54         return(stable[4],abs(n-k),choix_signe(k,n))
55     else :
56         return("element déjà stable")
57
```

On ne peut pas dire que le carbone et le silicium sont stables. On dit qu'ils ne forment pas d'ions monoatomiques

Une proposition en rapport avec le niveau 3 évoqué au départ. Une amélioration peut être apportée en mixant la détermination du symbole à partir du numéro atomique.

```
1 # Programme affichant la charge de l'ion monoatomique pouvant être formé
2 # par un atome
3
4 Z= int(input("Entrer le numéro atomique de l'atome"))
5 symbole=input("Entrer le symbole de l'élément chimique")
6
7 while (Z<1 or Z>18):
8     Z= int(input("Entrer un autre numéro atomique de l'atome"))
9
10 if (Z<=2): #Détermination du nombre d'électrons sur la couche externe
11     n_externe=Z
12 if (2<Z<=10):
13     n_externe=Z-2
14 if (10<Z<=18):
15     n_externe=Z-10
16
17 #détermination de la charge
18 if(4<n_externe<8 or Z==1 ): #Z==1 cas de l'hydrogène
19     charge= 8- n_externe
20     print(symbole,charge,"-")
21 if (0 <n_externe <4 and Z!=2 ): # Z!=2 cas de l'Helium
22     charge= n_externe
23     print(symbole,charge,"+")
24
25 if (n_externe==4 or n_externe==8 or Z==2):
26     print( "Cet atome ne forme pas d'ion")
--
```

Scenario 3 :

```
59
60 def symbole(a):
61     mendeleiv=["H","He","Li","Be","B","C","N",
62     nom=["Hydrogène","Hélium","Lithium","Béryl
63     b=mendeleiv.index(a)
64     c=b+1
65     return(nom[b],c)
66
```