

Exercice 1 : algorithme mystère

1. Le tableau de suivi de variables est :

	N	S
Ligne 2	43 015	-
Ligne 3	43 015	0
Ligne 4	Test vrai → 1.5	
Ligne 5	43 015	5
Ligne 6	4 301	5
Ligne 4	Test vrai → 1.5	
Ligne 5	4 301	6
Ligne 6	430	6
Ligne 4	Test vrai → 1.5	
Ligne 5	430	6
Ligne 6	43	6
Ligne 4	Test vrai → 1.5	
Ligne 5	43	9
Ligne 6	4	9
Ligne 4	Test vrai → 1.5	
Ligne 5	4	13
Ligne 6	0	13
Ligne 4	Test faux → 1.8	
	FIN	

2. Cet algorithme prend en entrée un nombre entier N et calcule la somme des chiffres contenus dans ce nombre.

Exercice 2 : compte bancaire

L'algorithme est similaire à celui du devoir maison numéro 1, les explications sont les mêmes. Il faut simplement faire attention aux cas particuliers.

```
solde = float(input("Quel est votre solde initial ? "))
montant = float(input("Quel montant souhaitez-vous transférer ? "))
operation = input("Quelle opération souhaitez-vous effectuer ?
    Pour un dépôt, tapez 1 ; pour un retrait, tapez 2. ")

if (montant < 0):
    print("Le montant à transférer doit être positif !")
else:
    if (operation == "1"):
        solde2 = solde + montant
        print("Votre nouveau solde est de", solde2, "euros.")
    elif (operation == "2"):
        solde2 = solde - montant
        if (solde2 < 0):
            print("Vous ne pouvez pas retirer autant d'argent.")
        else:
            print("Votre nouveau solde est de", solde2, "euros.")
    else:
        print("Opération non valide : vous devez taper 1 ou 2.")
```

Bien sûr comme dans le DM1, on pouvait faire rentrer les instructions dans une boucle pour être sûr que l'utilisateur tape quelque chose de correct.

Exercice 3 : déménagement

1. On peut rouler au maximum 250 km, or l'aller-retour entre l'entreprise et l'ancien appartement demande déjà 30 km (il faut bien aller de l'entreprise à l'appartement juste après avoir loué le camion pour faire le premier chargement, et il faut bien le rendre à la fin). Il reste donc 220 km, sur lesquels on peut donc faire au maximum 4 allers-retours de déménagement. On peut donc au maximum déménager 32 meubles en utilisant ce contrat.
2. Il y a 100€ de coûts fixes, auxquels il faut rajouter 0,1€ par kilomètre. Il y a 30 km en camion (aller-retour entreprise / ancien appartement) + 50 km en camion par aller-retour avec les meubles. On peut aussi compter 30 km de carburant en voiture aller-retour entreprise / ancien appartement donc 3€, à moins qu'il n'existe des transports en commun (qui sont peut-être gratuits comme à Châteauroux), car il faut bien se déplacer pour aller chercher le camion ! On va ici supposer qu'on va chercher le camion gratuitement (vélo par ex.)
 - (a) Lorsque $n = 5$, tout peut se faire en un chargement :
cela coûte $100€ + (30 + 50) \times 0,1€ = 108€$
 - (b) Lorsque $n = 8$, c'est identique : cela coûte 108€
 - (c) Lorsque $n = 19$, tout peut se faire en trois chargements :
cela coûte $100€ + (30 + 3 \times 50) \times 0,1€ = 118€$
 - (d) Lorsque $n = 25$, tout peut se faire en quatre chargements :
cela coûte $100€ + (30 + 4 \times 50) \times 0,1€ = 123€$
3. On déduit facilement que de 1 à 8 meubles, cela coûte 108€, de 9 à 16 cela coûte 113€, de 17 à 24 cela coûte 118€ et de 25 à 32 cela coûte 123€. On peut écrire le programme suivant :

```
n = int(input("Combien de meubles souhaitez-vous déménager ? "))

if (n > 32):
    print("Vous devez changer de contrat !")
else:
    if (n <= 8):
        prix = 108
    elif (n <= 16):
        prix = 113
    elif (n <= 24):
        prix = 118
    else:
        prix = 123
    print("Ce déménagement vous coûtera", prix, "euros.")
```

On peut aussi se rendre compte qu'il suffit de compter le nombre d'allers-retours ancien / nouvel appartement : il faut diviser n par 8 et arrondir à l'unité supérieure (l'arrondi à l'unité supérieure s'appelle le "plafond"), ce qui peut se faire en calculant le quotient $n//8$ et le reste $n\%8$. Si le reste est strictement positif, on ajoute alors 1 au quotient sinon c'est directement le quotient. On peut faire l'opération "arrondi au supérieur de la division par 8" en calculant $(n + 7)//8$.

```
n = int(input("Combien de meubles souhaitez-vous déménager ? "))

nb_ar = (n+7)//8
if (nb_ar > 4):
    print("Vous devez changer de contrat !")
else:
    prix = 103 + 5*nb_ar
    print("Ce déménagement vous coûtera", prix, "euros.")
```

Exercice 4 : indentation

Voici les trois manières de désambiguïser cet algorithme :

Indentation 1.

```
Variable :  
a est un nombre entier.  
  
Corps de l'algorithme :  
a prend la valeur 6  
Si a < 5  
    a prend la valeur a + 1  
    Si a < 7  
        a prend la valeur 2 × a  
    Sinon  
        a prend la valeur a + 3  
    Fin Si  
Fin Si  
  
Sortie  
afficher a
```

Indentation 2.

```
Variable :  
a est un nombre entier.  
  
Corps de l'algorithme :  
a prend la valeur 6  
Si a < 5  
    a prend la valeur a + 1  
    Si a < 7  
        a prend la valeur 2 × a  
    Fin Si  
Sinon  
    a prend la valeur a + 3  
Fin Si  
  
Sortie  
afficher a
```

Indentation 3.

```
Variable :  
a est un nombre entier.  
  
Corps de l'algorithme :  
a prend la valeur 6  
Si a < 5  
    a prend la valeur a + 1  
Fin Si  
Si a < 7  
    a prend la valeur 2 × a  
Sinon  
    a prend la valeur a + 3  
Fin Si  
  
Sortie  
afficher a
```

La première indentation fait afficher 6 : le premier test est faux, donc on sort directement.

La seconde indentation fait afficher 9 : le premier test est faux, donc on rentre dans le sinon qui ajoute 3.

La troisième indentation fait afficher 12 : le premier test est faux, mais il y a également le second test à effectuer qui est vrai, donc on multiplie par 2.

Exercice 4 : indentation

Voici les trois manières de désambiguïser cet algorithme :

Indentation 1.

```
Variable :  
a est un nombre entier.  
  
Corps de l'algorithme :  
a prend la valeur 6  
Si a < 5  
    a prend la valeur a + 1  
    Si a < 7  
        a prend la valeur 2 × a  
    Sinon  
        a prend la valeur a + 3  
    Fin Si  
Fin Si  
  
Sortie  
afficher a
```

Indentation 2.

```
Variable :  
a est un nombre entier.  
  
Corps de l'algorithme :  
a prend la valeur 6  
Si a < 5  
    a prend la valeur a + 1  
    Si a < 7  
        a prend la valeur 2 × a  
    Fin Si  
Sinon  
    a prend la valeur a + 3  
Fin Si  
  
Sortie  
afficher a
```

Indentation 3.

```
Variable :  
a est un nombre entier.  
  
Corps de l'algorithme :  
a prend la valeur 6  
Si a < 5  
    a prend la valeur a + 1  
Fin Si  
Si a < 7  
    a prend la valeur 2 × a  
Sinon  
    a prend la valeur a + 3  
Fin Si  
  
Sortie  
afficher a
```

La première indentation fait afficher 6 : le premier test est faux, donc on sort directement.

La seconde indentation fait afficher 9 : le premier test est faux, donc on rentre dans le sinon qui ajoute 3.

La troisième indentation fait afficher 12 : le premier test est faux, mais il y a également le second test à effectuer qui est vrai, donc on multiplie par 2.