

Exercices d'algorithmique : corrigés

Exercice 2

1. Dans l'algorithme suivant, préciser les entrées, les instructions et les sorties :

Ici une entrée : l'entier a et une sortie ; l'entier b .

2. Exécuter l'algorithme 2 après saisie de l'entier 6 pour la variable a en utilisant le tableau de mémoires ci-dessous :

Numéro d'instruction	a	b	c
1	6		
2	6	36	
3	6	36	72
4	6	36	72
5	6	39	72

Exercice 4

Par sécurité, un véhicule doit respecter une distance minimale avec le véhicule qui le précède, afin d'avoir le temps de freiner avant une collision. Ce temps correspond à celui de la perception puis de la réaction du conducteur, ainsi que des possibilités de freinage du véhicule.

Ce temps est fonction de la vitesse v du véhicule. Des études statistiques ont montré que cette distance D peut être calculée par la formule : $D = 8 + 0,2v + 0,003v^2$, où v est en km/h et D en mètres.

1. Calculer la distance de freinage pour une vitesse de 100 km/h.

Pour $v = 100 \text{ km/h}$, $D = 8 + 0,2v + 0,003v^2 = 8 + 20 + 30 = 58 \text{ m}$

2. Ecrire un algorithme qui affiche cette distance D en sortie lorsqu'est fournie la vitesse v en entrée.

D et v sont de type réel
Saisir v
 $D \leftarrow 8 + 0,2v + 0,003v^2$
Afficher D

3. Ci-dessous le programme Python correspondant :

```

9
10 v=float(input('v='))
11 D=8+0.2*v+0.003*v**2
12 print('Pour une vitesse de',v,'km/h, la distance de freinage est D=',D,'m')
```

Exercice 5 : indice de masse corporelle avec deux variables

3. Ecrire cet algorithme et son codage en langage Python.

Algorithme IMC	Programme en langage Python
Saisir M Saisir T $IMC \leftarrow M / T^2$ Afficher IMC	<pre> masse=float(input("saisissez votre masse en kg")) taille=float(input("saisissez votre taille en m")) IMC=masse/taille**2 print ("Votre IMC est : ",IMC)</pre>

4. Utiliser le programme pour obtenir l'IMC d'un individu de taille 1,70m et de poids 65kg

```

>>>
Votre IMC est : 22.49134948096886
>>>
```

Exercice 6 : coût d'un séjour avec deux variables

Une association organise un séjour balnéaire dans un hôtel. Le coût journalier par personne de la pension à l'hôtel est 80€ et celui du voyage est de 300€.

1. Quel est le coût total d'un séjour d'une semaine pour 10 personnes ?

Le coût sera $S = 10 \times (300 + 7 \times 80) = 8600$ €

2. Ecrire un algorithme qui affiche le coût total du séjour selon le nombre J de jours réservés et le nombre N de participants. Coder cet algorithme en langage Python.

Algorithme coût	Programme en langage Python
Saisir N Saisir J $S \leftarrow N \times (300 + J \times 80)$ Afficher « Le séjour coûtera S € »	<pre>N=int(input("saisissez le nombre de personnes")) J=float(input("saisissez le nombre de jours")) S=N*(300+J*80) print ("Le séjour de : ",N,"jours pour ",J,"jours coûtera",S,"euros")</pre>

Exercice 7

1. Ecrire un algorithme qui prend en entrées la largeur l et la longueur L d'un rectangle et qui, en sorties, donne l'aire, le périmètre et la longueur de la diagonale de ce rectangle.

l, L, A, P et D sont de type réel
Saisir l et L
 $A \leftarrow l \times L$
 $P \leftarrow 2(l + L)$
 $D \leftarrow \sqrt{l^2 + L^2}$
Afficher A, P et D

2. Traduire cet algorithme en langage Python avec des messages finaux adaptés à la situation.

```
8 from math import* #chargement du module "math"
9
10 #####Exercice 10 : aire, périmètre et diagonale d'un rectangle##
11
12 #####ENTREES#####
13 l=float(input('l='))
14 L=float(input('L='))
15 #####
16
17 #####TRAITEMENT#####
18 A=l*L#aire du rectangle
19 P=2*(l+L)#périmètre du rectangle
20 D=sqrt(l**2+L**2)#longueur de la diagonale du rectangle
21 #####
22
23 #####SORTIES#####
24 print ("L'aire du rectangle est A=",A )
25 print ('Le périmètre du rectangle est P=',P )
26 print ('La longueur de la diagonale est D=',D )
27
```

Exercice 9

1. *Juste la réponse* : cet algorithme prend trois réels en entrées et renvoie le plus petit des trois au sens large.

2. Programmer cet algorithme en langage Python. On affichera un message adapté à la situation.

```
14 A=float(input('A='))
15 B=float(input('B='))
16 C=float(input('C='))
17 m=A
18
19 if (B<=m):
20     m=B#m est le plus petit des nombres A et B
21
22 if (C<=m):#on compare m au troisième nombre
23     m=C
24
25 print('Le plus petit des trois nombres est : ',m)
```

Exercice 11

Une maison d'édition recrute des vendeurs à domicile pour placer des collections de livres d'art. Elle propose, au choix, deux types de contrats :

contrat 1 : salaire mensuel fixe de : 3150 € + 165 € par collection vendue.

contrat 2 : pas de salaire fixe, mais 240 € par collection vendue.

1. Ecrire un algorithme qui prend le nombre x de collections vendues en entrée et qui renvoie, en sorties, le numéro du contrat le plus intéressant puis le revenu qu'il permettra de gagner pour ces x collections vendues.

```
x, R1 et R2 sont de type entier
Saisir x
R1 ← 3150 + 65x
R2 ← 240x
Si R1 < R2 alors
    Afficher : « choisir le contrat 2 et vous gagnerez R2 euros »
sinon
    Afficher : « choisir le contrat 1 et vous gagnerez R1 euros »
Fin Si-sinon
```

2. Programmer cet algorithme en langage Python. On affichera les messages adaptés à la situation.

```
x=int(input('x='))
R_1=3150+165*x
R_2=240*x

if (R_1<=R_2):
    print ('Choisissez le contrat 1 et vous gagnerez ',R_1,'euros')
else:
    print ('Choisissez contrat 2 et vous gagnerez ',R_2,'euros')
```

Exercice 13 : IMC avec elif

1. Ecrire un programme Python que permet la saisie de la masse en kg et la taille en m d'un individu et de lui préciser dans quel cas de figure il se situe.

```
masse=float(input("saisissez votre masse en kg"))
taille=float(input("saisissez votre taille en m"))
IMC=masse/taille**2

if(IMC<=16.5):
    print ("Vous êtes en état de dénutrition")
elif(16.5<IMC<=18.5):
    print("Vous êtes en état de maigreur")
elif(18.5<IMC<=25):
    print("Votre poids est normal")
elif(25<IMC<=30):
    print("Vous êtes en surpoids")
elif(30<IMC<=35):
    print("Vous êtes en état d'obésité modérée")
elif(35<IMC<=40):
    print("Vous êtes en état d'obésité sévère")
else:
    print("Vous êtes en état d'obésité morbide")
```

Exercice 14 : fonction affine par morceaux

1. Ecrire un programme en langage Python qui permet la saisie du nombre d'hélices qu'Antoine souhaite acheter et qui affiche le prix qu'il aura à payer pour cet achat : voir ci-dessous pour le programme.

2. Tester ce programme sur les valeurs de q suivantes : 5, 20, 60 et 200.

Pour $q = 1$, on paye 10 €, pour $q = 20$, on paye 22 €, pour $q = 60$, on paye 51 € et pour $q = 200$, on paye 107 €.

```

q=int(input("saisissez le nombre d'hélices que vous souhaitez acheter"))

if(q<10):
    print ("Vous paierez",q*1+5,"€")
elif(10<=q<50):
    print("Vous paierez",q*0.8+6,"€")
elif(50<=q<100):
    print("Vous paierez",q*0.75+6,"€")
else:
    print("Vous paierez",q*0.5+7,"€")

```

Exercice 15

1. Recopier et compléter le programme python ci-dessous...

```

age=int(input('âge='))
if (age<=18 or age>=60):
    print ("L'entrée au musée est gratuite pour vous")
elif(age>18 and age<=25):
    print("L'entrée au musée vous coûtera 10€")
else:
    print("L'entrée au musée vous coûtera 20€")

```

2. Tester ce programme sur les valeurs 5 ans, 20 ans, 30 ans et 70 ans.

On trouve successivement : « gratuit », 10€, 20 € et « gratuit ».

Exercice 16

1.a. En complétant le tableau ci-dessous, déterminer les valeurs de S et de i à la fin du traitement.

Numéro d'instruction	S	i
<u>Initialisation</u>	0	1
1	1	2
2	3	3
3	6	4
4	10	5
5	15	6

2.a. Ecrire un algorithme qui prend un entier n en entrée et affiche, en sortie, la somme des entiers de 1 à n .

S , n et i sont de type entier Affecter à S la valeur 0 Pour i de 1 à n faire $S \leftarrow S + i$ fin Pour Afficher S

2.b. Ecrire cet algorithme en langage Python.

```

14 n=int(input('n='))
15
16 S=0 #on initialise S
17 for i in range(n): #ici, on aurait pu aussi écrire : "for i in range(1,n+1)"
18     S=S+i
19 print('La somme des',n,'premiers entiers est S =',S)

```

3.a. Ecrire un algorithme qui prend un entier naturel n en entrée et affiche, en sortie, le produit des entiers de 1 à n .

P , n et i sont de type entier
Affecter à P la valeur 1
Pour i de 1 à n faire
$P \leftarrow P \times i$
fin Pour
Afficher P

3.b. Traduire cet algorithme en langage Python.

```

11 n=int(input('n='))
12
13 P=1 #on initialise P la variable où sera stocké le produit
14 for i in range(2,n+1): #ici, on aurait pu aussi écrire : "for i in range(1,n+1)"
15     P=P*i
16 print('Le produit des',n,'premiers entiers est P =',P)

```

Exercice 17

1. Ecrire un algorithme qui prend un entier N en entrée et qui renvoie, en sortie, la table de N .

N et i sont de type entier
Pour i de 0 à N faire
Afficher $N \times i$
fin Pour

2. Traduire cet algorithme en langage Python.

```

N=int(input('Saisir la table choisie N='))

table=""#initialisation de la chaîne dans laquelle on stockera la table de N

for i in range (11):

    print(N,'x',i,"=",N*i)

```

Exercice 18

Un produit se vend 300 € en 2014. On considère que son prix augmente de 2% par an.

- En 2015 son prix sera $P_1 = 300 + \frac{2}{100} \times 300 = 306$ €.
- En 2016 son prix sera $P_2 = 306 + \frac{2}{100} \times 306 = 312,12$ €.

1. Calculer le prix P_3 du produit en 2017 puis le prix P_4 en 2018 au centime près.

$$P_3 = 312,12 + \frac{2}{100} \times 312,12 = 318,3624 \text{ € et } P_4 = 318,3624 + \frac{2}{100} \times 318,3624 = 324,729648 \text{ €}.$$

2. Ecrire un algorithme qui prend en entrée le nombre N d'années après 2014 et donne, en sortie, le prix du produit en l'année $2014 + N$.

N et i sont de type entier
P est de type réel
Affecter à P la valeur 300
Pour i de 1 à N faire
$P \leftarrow P + 0,02 \times P$
fin Pour
Afficher P

3.a. Ecrire cet algorithme en langage Python.

```

13
14 N=int(input('N='))
15 P=300#initialisation
16
17 for i in range(N): #ici, on aurait pu aussi écrire : "for i in range(1,n+)"
18     P=P*1.02 #augmenter un prix de 2% c'est le multiplier par 1,02
19 print("Le prix l'année",2014+N,"=",P)

```

3.b. Que suffirait-il de modifier dans ce programme pour que s'affichent les prix du produit chaque année entre les années 2015 et 2014 + N .

- Afin d'obtenir le prix chaque année, plutôt que le prix la dernière année uniquement, il suffit de demander l'affichage des prix à l'intérieur de la boucle **Pour**.
- Remarquer que l'on peut même indiquer l'année courante. Essayer de comprendre pourquoi l'année courante est 2014 + i + 1 .

```

13
14 N=int(input('N='))
15 P=300#initialisation
16
17 for i in range(N): #ici, on aurait pu aussi écrire : "for i in range(1,n+)"
18     P=P*1.02 #augmenter un prix de 2% c'est le multiplier par 1,02
19     print("Le prix l'année",2014+i+1,"=",P)

```

Exercice 20 : nombre de 6

1. Ecrire un algorithme prenant en entrée un entier N représentant le nombre de lancers d'un dé et affichant le nombre de 6 obtenus ainsi que la fréquence d'apparition des numéros 6 dans l'échantillon.

```

N, S et i sont de type entier
Saisir N
S ← 0
Pour i de 1 à N faire
    d = aléa(1,6)
    Si d = .... alors
        S ← .....
    Fin si
fin Pour
Afficher .....

```

2. Traduire cet algorithme en langage Python.

```

#####
from math import *
from random import *
#####

N=int(input("N="))
S=0#S est la variable dénombrant les apparitions du nombre 6

for i in range(N):
    d=randint(1,6)#on fait rouler le dé
    if d==6:
        S=S+1
f=S/N#fréquence d'apparitions du numéro 6

print("Le numéro 6 est apparu ",S," fois soit une fréquence d'apparition de ",f)

```

Exercice 22 : boucle tant que

En prévision d'une course de vélo, Fanny suit le programme d'entraînement suivant tous les samedis : elle parcourt 30 km le premier samedi, puis augmente chaque samedi de 5 km la distance parcourue. Elle arrêtera cet entraînement lorsque qu'elle aura parcouru et cumulé une distance totale supérieure à 1000 km.

1. Calculer la distance D parcourue le deuxième samedi et la distance totale T parcourue au bout de deux samedis d'entraînement.

La distance parcourue le second samedi : $D_2 = 30 + 5 = 35 \text{ km}$ et $T = D_1 + D_2 = 30 + 35 = 65 \text{ km}$.

2. Nous allons écrire un algorithme afin de connaître le nombre de samedis nécessaires à l'entraînement de Fanny. Préciser les variables qui seront utiles à l'écriture de cet algorithme ?

Nous utiliserons trois variables : D , N et T . Dans D on stockera la distance parcourue le samedi courant, dans N le nombre de samedis et dans T la distance totale.

3. Compléter l'algorithme 1 ci-dessous afin que qu'il affiche le nombre de samedis nécessaires à l'entraînement de Fanny.

```
D ← 30
T ← 30
N ← 1
Tant que  $T \leq 1000$  . faire
    D ← D + 5
    T ← T + D
    N ← N + 1
fin Tq
Afficher N
```

4. Traduire cet algorithme en langage Python et donner le nombre de samedis nécessaires à l'accomplissement de l'entraînement de Fanny.

```
D=30
T=30
N=1

while (T<1000):
    D=D+5
    T=T+D
    N=N+1

print("Il faudra ",N," samedis à Fanny pour accomplir son entraînement ")

| Il faudra 16 samedis à Fanny pour accomplir son entraînement
| >>>
```

5. Modifier et programmer le programme afin qu'il affiche aussi la distance totale parcourue et la distance du dernier samedi.

```
D=30
T=30
N=1

while (T<1000):
    D=D+5
    T=T+D
    N=N+1

print("Il faudra ",N," samedis à Fanny pour accomplir son entraînement ")
print ("Fanny aura parcouru un total de ",T,"km dont ",D," km le dernier samedi")

| >>>
| Il faudra 16 samedis à Fanny pour accomplir son entraînement
| Fanny aura parcouru un total de 1080 km dont 105 km le dernier samedi
| >>>
```

Exercice 23

On considère l'algorithme suivant :

```
S et i sont de type entier
Affecter à S la valeur 0
Affecter à i la valeur 1
Tant que  $i \leq 10$  faire
    S ← S + i
    i ← i + 2
Fin Tant que
Afficher S
```

1. En complétant le tableau ci-dessous, déterminer les valeurs de S et de i à la fin du traitement.

Numéro d'instruction	S	i
<u>Initialisation</u>	0	1
1	1	3
2	4	5
3	9	7
4	16	9
5	25	11

2.a. Ecrire un algorithme qui prend un entier N en entrée et affiche, en sortie, la somme des entiers **impairs** entre 1 et N .

```

S, N et i sont de type entier
S ← 0
i ← 1
Saisir N
Tant que i ≤ N faire
    S ← S + i
    i ← i + 2
Fin Tant que
Afficher S

```

2.b. Traduire cet algorithme en langage Python.

Le programme commenté

```

10
11 S=0#initialisation du nombre de la somme des entiers impairs
12 i=1#initialisation du premier entier impair
13
14 N=int(input('N='))#saisie de la borne supérieure de la boucle Tant que
15
16 while (i<=N):#le nombre impair courant n'a pas dépassé N
17     S=S+i
18     i=i+2
19
20 print ('la somme des entiers impairs de 1 à',i-2,'est S=',S)#à la dernière entrée dans la boucle,
21                                     #i est incrémenté de 2 mais ce nouvel entier
22                                     #dépasse N, il ne figurera donc pas dans la
23                                     #somme. Le dernier entier de la somme est
24                                     #donc i-2

```

Un affichage possible

```

N=10
la somme des entiers impairs de 1 à 9 est S= 25

```

Exercice 24

Une ville de 10 000 habitants en 2014 voit sa population augmenter de 3% par an.

1.a. Montrer que chaque année sa population est multipliée par 1,03.

Si on appelle P_n le nombre d'habitants l'année n , on a : $P_{n+1} = P_n + \frac{3}{100} P_n = \left(1 + \frac{3}{100}\right) P_n = 1,03 \times P_n$. La population est bien multipliée par 1,03 de l'année n à l'année $n+1$.

1.b. Calculer, à l'habitant près, le nombre d'habitants en 2015, 2016 et 2017.

$P_1 = 1,03 \times 10000 = 10300$, $P_2 = 1,03 \times 10300 = 10609$ et $P_3 = 1,03 \times 10609 \approx 10927$ à 1 habitant près.

2. Ecrire un algorithme qui renvoie, en sortie, le nombre d'années au bout duquel la population dépasse 13 400 habitants.


```

P est de type réel et n de type
entier
P ← 10000
n ← 0
Tant que P ≤ 13400 faire
    P ← 1,03 × P
    n ← n + 1
Fin Tant que
Afficher n

```

3. Traduire cet algorithme en langage Python.

```

1 P=10000#initialisation de la variable P représentant la population
2 n=0#initialisation de la variable N représentant le nombre d'années
3
4 while (P<=13400):#vérification du seuil
5     P=1.03*P#mise à jour de la population
6     n=n+1#incrémentatation du nombre d'années
7
8 print ('la population dépasse 13400 individus en',2014+n)
9

```

Exercice 26 : fonction

1. Programmer une fonction **soin** permettant d'obtenir la quantité de médicament suffisante pour calmer un individu connaissant sa taille et sa masse. La fonction **surface** de l'exercice 25 sera appelée dans ce programme.

```

def surface(L,M):
    S=sqrt(L*M)/6
    return(S)

def soin(L,M):
    return(5*surface(L,M))

```

2. Un individu brûlé se présente aux urgences. Il mesure 1,90 m et pèse 92 kg.
Quelle quantité de médicament sera-t-il nécessaire pour soigner ce patient ?

```

>>>
>>> soin(1.9,92)
11.017662587358938
>>>

```

Exercice 27

1. Programmer une fonction, nommée **maximum**, qui prend trois réels en arguments et renvoie le plus grand des trois ;

```

11
12 def maximum(a,b,c):#trois réels en arguments
13     M=a
14     if (b>M):
15         M=b
16     if (c>M):
17         M=c
18     return(M)#le plus grand des trois est renvoyé
19

```

2. Programmer une fonction, nommée **minimum**, qui prend trois réels en arguments et renvoie le plus petit des trois.

```

20
21 def minimum(a,b,c):#trois réels en arguments
22     m=a
23     if (b<m):
24         m=b
25     if (c<m):
26         m=c
27     return(m)#le plus petits des trois est renvoyé
28

```

3. Ecrire un programme qui permet la saisie de trois réels et affiche le plus grand des trois et le plus petit des trois.

```

31
32 a=float(input("saisir un réel a : "))
33 b=float(input("saisir un réel b : "))
34 c=float(input("saisir un réel c : "))
35 print(" ")#saut de ligne
36 M=maximum(a,b,c)
37 m=minimum(a,b,c)
38 print(M,"est la plus grand et ", m, " est le plus petit des nombres " ,a," , ",b," et ",c)
39

```

Exercice 28

1. Programme une fonction nommée **fact** qui prend un entier $n \geq 1$ en argument et retourne le produit des entiers de 1 à n .

```

from math import*

def fact(n):#fact prend un entier naturel n en argument
    P=1
    for i in range(1,n+1):
        P=P*i
    return P

```

2. Ecrire un programme qui prend un entier $N \geq 1$ en entrée et qui, pour tout $1 \leq i \leq N$, affiche le produit des entiers de 1 à i .

```

#####Programme principal#####
N=int(input("Saisir N ="))
print(" ")#saut de ligne

for i in range(1,N+1):
    print(i,"!=" ,fact(i))
    print(" ")#saut de ligne

```

3. Que remarque-t-on pour l'entrée $N = 30$?

Voici les affichages obtenus pour l'entrée $N = 30$:

Saisir N =30	Pour i = 11 : 39916800	Pour i = 23 : 25852016738884976640000
Pour i = 1 : 1	Pour i = 12 : 479001600	Pour i = 24 : 620448401733239439360000
Pour i = 2 : 2	Pour i = 13 : 6227020800	Pour i = 25 : 15511210043330985984000000
Pour i = 3 : 6	Pour i = 14 : 87178291200	Pour i = 26 : 403291461126605635584000000
Pour i = 4 : 24	Pour i = 15 : 1307674368000	Pour i = 27 : 10888869450418352160768000000
Pour i = 5 : 120	Pour i = 16 : 20922789888000	Pour i = 28 : 3048883446117138605015040000000
Pour i = 6 : 720	Pour i = 17 : 355687428096000	Pour i = 29 : 88417619937397019545436160000000
Pour i = 7 : 5040	Pour i = 18 : 6402373705728000	Pour i = 30 : 2652528598121910586363084800000000
Pour i = 8 : 40320	Pour i = 19 : 121645100408832000	
Pour i = 9 : 362880	Pour i = 20 : 2432902008176640000	
	Pour i = 21 : 51090942171709440000	

Rmq : on constate évidemment la croissance extrêmement rapide de la suite des entiers i !.