

TS TP 3 (ALGORITHMIQUE) : Éléments de correction.

I) 2)

Programme codé sur AlgoBox :

CODE DE L'ALGORITHME :

```

1  VARIABLES
2  n EST_DU_TYPE NOMBRE
3  somme EST_DU_TYPE NOMBRE
4  DEBUT_ALGORITHME
5  POUR n ALLANT_DE 1 A 10
6  DEBUT_POUR
7  somme PREND_LA_VALEUR somme+n
8  FIN_POUR
9  AFFICHER somme
10 FIN_ALGORITHME

```

RÉSULTATS :

```

***Algorithme lancé***
55
***Algorithme terminé***

```

Généré par AlgoBox

I) 3) Exercice 1 :

a) Pour $N = 3$, on obtient 6 ; pour $N = 5$, on obtient 120.

b) Cet algorithme affiche le résultat du produit des N premiers entiers (non nuls), N étant l'entier saisi, c'est-à-dire $N!$

c) Algorithme codé :

Variables entières : N ; i et produit ($N > 0$)

Début de l'algorithme

Saisir N

produit $\leftarrow 1$

Pour i variant de 1 à N

 produit \leftarrow produit $\times i$

Fin Pour

Afficher produit

Fin de l'algorithme

Programme codé sur AlgoBox :

CODE DE L'ALGORITHME :

```

1  VARIABLES
2  N EST_DU_TYPE NOMBRE
3  i EST_DU_TYPE NOMBRE
4  produit EST_DU_TYPE NOMBRE
5  DEBUT_ALGORITHME
6  LIRE N
7  produit PREND_LA_VALEUR 1
8  POUR i ALLANT_DE 1 A N
9  DEBUT_POUR
10 produit PREND_LA_VALEUR produit*i
11 FIN_POUR
12 AFFICHER "La valeur de N est "
13 AFFICHER N
14 AFFICHER "La valeur de produit est "
15 AFFICHER produit
16 FIN_ALGORITHME

```

RÉSULTATS :

```

***Algorithme lancé***
La valeur de N est 5
La valeur de produit est 120
***Algorithme terminé***

```

Exercice 2 :a) *Version modifiée (codée) :*Variables : n ; i et somme (entiers)

Début de l'algorithme

Saisir n **Pour** i variant de 1 à n somme \leftarrow somme + i **Fin Pour**

Afficher somme

Fin de l'algorithme

b) Programme codé sur AlgoBox :

CODE DE L'ALGORITHME :

```

1  VARIABLES
2    n EST_DU_TYPE NOMBRE
3    somme EST_DU_TYPE NOMBRE
4    i EST_DU_TYPE NOMBRE
5  DEBUT_ALGORITHME
6    LIRE n
7    POUR i ALLANT_DE 1 A n
8      DEBUT_POUR
9        somme PREND_LA_VALEUR somme+i
10     FIN_POUR
11    AFFICHER "La valeur de n est "
12    AFFICHER n
13    AFFICHER "La valeur de somme est "
14    AFFICHER somme
15  FIN_ALGORITHME

```

RÉSULTATS :

```

***Algorithme lancé***
La valeur de n est 100
La valeur de somme est 5050
***Algorithme terminé***

```

II) 2)

CODE DE L'ALGORITHME :

```

1  VARIABLES
2    N EST_DU_TYPE NOMBRE
3    S EST_DU_TYPE NOMBRE
4  DEBUT_ALGORITHME
5    S PREND_LA_VALEUR 5000
6    TANT_QUE (S<8000) FAIRE
7      DEBUT_TANT_QUE
8        S PREND_LA_VALEUR 1.02*S
9        N PREND_LA_VALEUR N+1
10     FIN_TANT_QUE
11    AFFICHER N
12    AFFICHER S
13  FIN_ALGORITHME

```

RÉSULTATS :

```

***Algorithme lancé***
24
8042.1862
***Algorithme terminé***

```

II) 3)

Exercice 3 : On donne l'algorithme suivant :

Variables entières : n et r
 Début de l'algorithme
 Entrer n
 r prend la valeur n
Tant que $r \geq 11$
 r prend la valeur $r - 11$
Fin Tant que
 Afficher r
 Fin de l'algorithme

- a) Pour $n = 3$, on trouve $r = 3$ (la condition n'est pas réalisée dès le départ) ;
 Pour $n = 65$, on obtient $r = 10$;
 Pour $n = 121$, on trouve $r = 0$.
- b) Cet algorithme affiche le reste dans la division euclidienne de n par 11.

c) Programme codé sur AlgoBox :

```

1  VARIABLES
2    n EST_DU_TYPE NOMBRE
3    r EST_DU_TYPE NOMBRE
4  DEBUT_ALGORITHME
5    LIRE n
6    r PREND_LA_VALEUR n
7    TANT_QUE (r>=11) FAIRE
8      DEBUT_TANT_QUE
9        r PREND_LA_VALEUR r-11
10     FIN_TANT_QUE
11    AFFICHER r
12  FIN_ALGORITHME

```

Exercice 6 :Étape 1 :

a)

n	0	1	2	3	4	5	6	8	10	16
$f(n)$	0	4	1	10	2	16	3	4	5	8

b) Programme codé sur AlgoBox :

```

1  VARIABLES
2  n EST_DU_TYPE NOMBRE
3  y EST_DU_TYPE NOMBRE
4  DEBUT_ALGORITHME
5  LIRE n
6  SI (n%2==0) ALORS
7  DEBUT_SI
8  y PREND_LA_VALEUR n/2
9  FIN_SI
10 SINON
11 DEBUT_SINON
12 y PREND_LA_VALEUR 3*n+1
13 FIN_SINON
14 AFFICHER y
15 FIN_ALGORITHME

```

c) On retrouve les résultats de la question a) .

Étape 2 :

EXEMPLE : En utilisant les résultats du tableau de l'étape 1, vérifier les suites ci-dessous :

□ Nombre de départ : $n = 2$ 2 ; 1 car $f(2) = 1$.

La durée du vol est 1 car il faut une étape pour obtenir 1.

□ Nombre de départ : $n = 3$

3 ; 10 ; 5 ; 16 ; 8 ; 4 ; 2 ; 1

La durée du vol est 7.

 $f(3) = 10 ; f(10) = 5 ; f(5) = 16 ; f(16) = 8 ; f(8) = 4 ; f(4) = 2 ; f(2) = 1$.□ Nombre de départ : $n = 4$

4 ; 2 ; 1

La durée du vol est 2.

 $f(4) = 2 ; f(2) = 1$.□ Nombre de départ : $n = 5$

5 ; 16 ; 8 ; 4 ; 2 ; 1

La durée du vol est 5.

 $f(5) = 16 ; f(16) = 8 ; f(8) = 4 ; f(4) = 2 ; f(2) = 1$.□ Nombre de départ : $n = 6$

6 ; 3 ; 10 ; 5 ; 16 ; 8 ; 4 ; 2 ; 1

La durée du vol est 8.

 $f(6) = 3 ; f(3) = 10 ; f(10) = 5 ; f(5) = 16 ; f(16) = 8 ; f(8) = 4 ; f(4) = 2 ; f(2) = 1$.**a) Algorithme :**Variables : n et y

Début de l'algorithme

Entrer n Afficher n y prend la valeur de n **Tant que** $y > 1$ **Si** n est pair **Alors** y prend la valeur $\frac{n}{2}$ **Sinon** y prend la valeur $3n + 1$ **FinSi** n prend la valeur y Afficher n **Fin Tant que**

Fin de l'algorithme

Programme codé sur AlgoBox :

```

1  VARIABLES
2    n EST_DU_TYPE NOMBRE
3    y EST_DU_TYPE NOMBRE
4  DEBUT_ALGORITHME
5    LIRE n
6    AFFICHER n
7    y PREND_LA_VALEUR n
8    TANT_QUE (y>1) FAIRE
9      DEBUT_TANT_QUE
10     SI (n%2==0) ALORS
11       DEBUT_SI
12         y PREND_LA_VALEUR n/2
13       FIN_SI
14     SINON
15       DEBUT_SINON
16         y PREND_LA_VALEUR 3*n+1
17       FIN_SINON
18     n PREND_LA_VALEUR y
19     AFFICHER n
20     FIN_TANT_QUE
21  FIN_ALGORITHME

```

Programme codé « allégé » sur AlgoBox :

```

1  VARIABLES
2    n EST_DU_TYPE NOMBRE
3  DEBUT_ALGORITHME
4    LIRE n
5    AFFICHER n
6    TANT_QUE (n>1) FAIRE
7      DEBUT_TANT_QUE
8        SI (n%2==0) ALORS
9          DEBUT_SI
10           n PREND_LA_VALEUR n/2
11         FIN_SI
12       SINON
13         DEBUT_SINON
14           n PREND_LA_VALEUR 3*n+1
15         FIN_SINON
16       AFFICHER n
17       FIN_TANT_QUE
18  FIN_ALGORITHME

```

b) Algorithme :Variables : n ; y et nb

Début de l'algorithme

Entrer n Afficher n y prend la valeur n **Tant que** $y > 1$ **Si** n est pair **Alors** y prend la valeur $\frac{n}{2}$ **Sinon** y prend la valeur $3n + 1$ **FinSi** n prend la valeur y nb prend la valeur $nb + 1$ Afficher n (avec un retour ligne)**Fin Tant que**Afficher nb

Fin de l'algorithme

Programme codé sur AlgoBox :

```

1  VARIABLES
2  n EST_DU_TYPE NOMBRE
3  y EST_DU_TYPE NOMBRE
4  nb EST_DU_TYPE NOMBRE
5  DEBUT_ALGORITHME
6  LIRE n
7  AFFICHER n
8  y PREND_LA_VALEUR n
9  TANT_QUE (y>1) FAIRE
10 DEBUT_TANT_QUE
11 SI (n%2==0) ALORS
12 DEBUT_SI
13 y PREND_LA_VALEUR n/2
14 FIN_SI
15 SINON
16 DEBUT_SINON
17 y PREND_LA_VALEUR 3*n+1
18 FIN_SINON
19 n PREND_LA_VALEUR y
20 nb PREND_LA_VALEUR nb+1
21 AFFICHER n
22 FIN_TANT_QUE
23 AFFICHER "La durée du vol est  "
24 AFFICHER nb
25 FIN_ALGORITHME

```

Programme codé « allégé » sur AlgoBox :

```
1  VARIABLES
2    n EST_DU_TYPE NOMBRE
3    nb EST_DU_TYPE NOMBRE
4  DEBUT_ALGORITHME
5    LIRE n
6    AFFICHER n
7    TANT_QUE (n>1) FAIRE
8      DEBUT_TANT_QUE
9        SI (n%2==0) ALORS
10       DEBUT_SI
11         n PREND_LA_VALEUR n/2
12       FIN_SI
13     SINON
14       DEBUT_SINON
15         n PREND_LA_VALEUR 3*n+1
16       FIN_SINON
17     nb PREND_LA_VALEUR nb+1
18     AFFICHER n
19     FIN_TANT_QUE
20   AFFICHER "La durée du vol est  "
21   AFFICHER nb
22 FIN_ALGORITHME
```