

Série d'exercice N3 pour Module d'Info2

M.LICHOURI

02 Mai 2013

Dernière mise à jour : 2013/05/24 à 14:58:33

1 Série d'exercice 3 : Les Structures de Contrôle

1.1 Exercice 1

Ecrire un algorithme qui détermine la valeur de deux nombres connaissant leur somme et leur différence. $x + y = s(I)$

$$x - y = d(II)$$

$$(I) + (II) \Rightarrow 2x = s + d \Rightarrow x = (s + d)/2$$

$$(I) - (II) \Rightarrow 2y = s - d \Rightarrow y = (s - d)/2$$

Exercice 1

```
Algorithme EX1;  
1 VARIABLES  
2 x,y,s,d:entier;  
3 DEBUT  
4 Ecrire "Donnez les valeurs de s et d:"  
5 Lire(s,d);  
6 x <-- (s+d)/2;  
4 y <-- (s-d)/2;  
5 Ecrire("la valeur de x vaut: ",x);  
6 Ecrire("la valeur de y vaut: ",y);  
7 FIN.
```

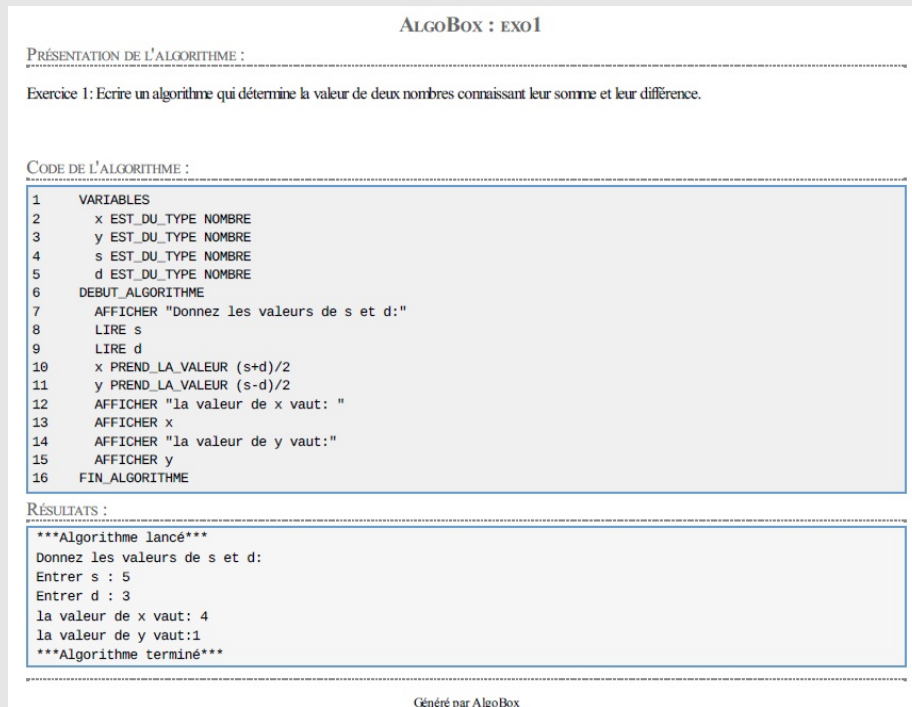


FIGURE 1 – Execution dans AlgoBox

1.2 Exercice 2

Ecrire un algorithme permettant de saisir l'abscisse d'un point A et de calculer son ordonnée $y = 2x^3 - 3x^2 + 4$

Exercice 2

```
Algorithme EX2;
1 VARIABLES
2 x,y:réel;
3 DEBUT
4 Ecrire("Donner la valeur de x:");
5 Lire(x);
6 y <-- 2*x^3 -3*x^2+4;
7 Ecrire("La valeur de y vaut: ",y);
8 FIN.
```

```
ALGOBox : EXO2

PRÉSENTATION DE L'ALGORITHME :
=====
Exercice 2
Ecrire un algorithme permettant de saisir l'abscisse d'un point A et de calculer son ordonnée
y=2x^3-3x^2+4

CODE DE L'ALGORITHME :
=====
1  VARIABLES
2  x EST_DU_TYPE NOMBRE
3  y EST_DU_TYPE NOMBRE
4  DEBUT_ALGORITHME
5  AFFICHER "Donner la valeur de x:"
6  LIRE x
7  y PREND_LA_VALEUR 2*pow(x,3)-3*pow(x,2)+4
8  AFFICHER "La valeur de y vaut: "
9  AFFICHER y
10 FIN_ALGORITHME

RÉSULTATS :
=====
***Algorithme lancé***
Donner la valeur de x:
Entrer x : 1
La valeur de y vaut: 3
***Algorithme terminé***

=====
Généré par AlgoBox
```

FIGURE 2 – Execution dans AlgoBox

1.3 Exercice 3

Déroulement d'un programme

1. $A=5$ $b=10$ $c=20$: $(a < b) \Rightarrow a=2*b=20 \Rightarrow x=a+b+c=50$
2. $A=15$ $b=10$ $c=20$: $(a > b \& a < c) \Rightarrow b=b+c=30 \Rightarrow x=65$
3. $A=15$ $b=10$ $c=10$: $(a > b \& a > c) \Rightarrow c=c+10=20$ et $b=b+c=30 \Rightarrow x=65$

1.4 Exercice 4

Décrire l'état de l'eau selon sa température

Exercice 4

```
Algorithme Temperature1;
1  VARIABLES
2    T : réels;
3  DEBUT
4    ECRIRE("Donnez la valeur de la température T: ");
5    LIRE(T);
6    SI (T<0) ALORS
7      ECRIRE("Son etat est: GLACE")
8    SINON
9      SI (T<100) ALORS
10     ECRIRE("Son etat est: EAU")
11   SINON
12     ECRIRE("Son etat est: VAPEUR")
13   FSI;
14 FSI;
15 FIN.
```

```
Algorithme Temperature2;
1  VARIABLES
2    T : réels;
3  DEBUT
4    ECRIRE("Donnez la valeur de la température T: ");
5    LIRE(T);
6    SI (T<0) ALORS
7      ECRIRE("Son etat est: GLACE")
8    FSI;
9    SI (T>0 et T<100) ALORS
10     ECRIRE("Son etat est: EAU")
11   FSI;
12   SI (T>100) ALORS
13     ECRIRE("Son etat est: VAPEUR")
14   FSI;
15 FIN.
```

RÉSULTATS :

```
***Algorithme lancé***
Donnez la valeur de la température T:
Entrer T : 50
Son etat est: EAU
***Algorithme terminé***
```

Généré par AlgoBox

FIGURE 3 – Résultat affiché dans l'exercice 4

1.5 Exercice 5

Calculer la somme ou le produit de deux nombres selon le choix entré.

Exercice 5

```
Algorithme EX5;
1 Variables
2 a,b,s,p : entiers;
3 choix: caractère;
4 Début
5 Ecrire('Donner les valeurs de a et b: ');
6 Lire(a,b);
7 Ecrire('Indiquer votre choix: ');
8 Lire(choix);
9 Si(choix=='s') alors
10   début
11     s <-- a+b ;
12     Ecrire('s= ',s)
13   fin;
14 Sinon Si(choix=='p') alors
15   début
16     p <-- a*b;
17     Ecrire ('p= ',p);
18   fin;
19   Sinon Ecrire('Erreur')
20   Fsi;
21 Fsi;
22 Fin.
```

RÉSULTATS :

```
***Algorithme lancé***
Donnez les valeurs de x et y:
Entrer x : 2
Entrer y : 5
Indiquez votre choix
Entrer choix : s
la somme de x et y est:7
***Algorithme terminé***
```

Généré par AlgoBox

FIGURE 4 – Résultat affiché dans l'exercice 5(Somme)

```
RÉSULTATS :
***Algorithme lancé***
Donnez les valeurs de x et y:
Entrer x : 2
Entrer y : 5
Indiquez votre choix
Entrer choix : p
le produit de x et y est: 10
***Algorithme terminé***

Généré par AlgoBox
```

FIGURE 5 – Résultat affiché dans l'exercice 5(Produit)

1.6 Exercice 6

Ecrire un algorithme qui indique si trois nombres x , y et z dans cet ordre forment une progression arithmétique.

Exercice 6

```
1 Algorithme EX6;
2 Variables x,y,z : entiers;
3 Début
4 Ecrire('Donner les valeurs de x,y et z: ');
5 Lire(x,y,z);
6 Si(y-x==z-y) alors Ecrire('Ces trois nombres forment une Suite');
7 Sinon Ecrire('Non');
8 Fsi;
9 Fin.
```

```
RÉSULTATS :
***Algorithme lancé***
Donnez les valeurs de x, y et z:
Entrer x : 2
Entrer y : 5
Entrer z : 8
Ces trois nombre forment une progression arithmétique
***Algorithme terminé***

Généré par AlgoBox
```

FIGURE 6 – Résultat affiché dans l'exercice 6

1.7 Exercice 7

Ecrire un algorithme qui range trois nombres entiers dans l'ordre croissant. L'affichage se fera de la manière suivante :

Exercice 7 partie 1

```
1 Algorithme EX7;
2   VARIABLES
3     x,y,z:entier;
4   DEBUT
5     Ecrire("Donnez les valeurs de x,y et z:");
6     Lire(x,y,z);
7     SI (x>=y et x>=z) ALORS
8       DEBUT
9         SI (y>=z) ALORS
10          DEBUT
11            Ecrire(" Le plus grand est: ",x);
12            Ecrire(" Le moyen est: ",y);
13            Ecrire(" Le plus petit est: ",z);
14          FIN;
15        SINON
16          DEBUT
17            Ecrire(" Le plus grand est: ",x);
18            Ecrire(" Le moyen est: ",z);
19            Ecrire(" Le plus petit est: ",y);
20          FIN;
21        FSI;
22      FIN;
23    FSI;
24    SI (y>=x et y>=z) ALORS
25      DEBUT
26        SI (x>=z) ALORS
27          DEBUT
28            Ecrire(" Le plus grand est: ",y);
29            Ecrire(" Le moyen est: ",x);
30            Ecrire(" Le plus petit est: ",z);
31          FIN;
32        SINON
33          DEBUT
34            Ecrire(" Le plus grand est: ",y);
35            Ecrire(" Le moyen est: ",z);
36            Ecrire(" Le plus petit est: ",x);
37          FIN;
38        FINSI;
39      FIN;
40    FSI;
```

Exercice 7 partie 2

```
41  SI (z>=x et z>=y) ALORS
42      DEBUT
43      SI (x>=y) ALORS
44          DEBUT
45              Ecrire(" Le plus grand est: ",z);
46              Ecrire(" Le moyen est: ",x);
47              Ecrire(" Le plus petit est: ",y);
48          FIN;
49      SINON
50          DEBUT
51              Ecrire(" Le plus grand est: ",z);
52              Ecrire(" Le moyen est: ",y);
53              Ecrire(" Le plus petit est: ",x);
54          FIN;
55      FINSI;
56      FIN;
57  FSI;
58  FIN.
```

RÉSULTATS :

```
***Algorithme lancé***
Donnez les valeurs de x,y et z:
Entrer x : 3
Entrer y : 1
Entrer z : 2
Le plus grand est3
Le moyen est2
Le plus petit est1
***Algorithme terminé***
```

Généré par AlgoBox

FIGURE 7 – Résultat affiché dans l'exercice 7

1.8 Exercice 8

Ecrire un algorithme qui calcule la somme des 20 premiers nombres pairs.

Exercice 8

```
1 Algorithme EX8;
2 VARIABLES
3   i,x,som:entier;
4 DEBUT
5   x <-- 0;
6   som <-- 0;
7   POUR i <-- 1 A 20 Faire
8     x <-- x+2;
9     som <-- som+x;
10  Fait;
11  Ecrire("La somme vaut: ",som);
12 FIN.
```

RÉSULTATS :

```
***Algorithme lancé***
La somme vaut: 420
***Algorithme terminé***
```

Généré par AlgoBox

FIGURE 8 – Résultat affiché dans l'exercice 8

1.9 Exercice 9

Ecrire un algorithme qui calcule $n!$ de manière descendante.

Exercice 9

```
1 Algorithme EX9_Factoriel;
2 VARIABLES
3   i,n,fact:entier;
4 DEBUT
5   fact <-- 1;
6   Ecrire("Donner la valeur de n: ");
7   Lire(n);
8   POUR i <-- 1 A n Faire
9     fact <-- fact*i;
10  Fait;
11  Ecrire(n,"!=" ,fact);
12 FIN.
```

```
RÉSULTATS :
***Algorithme lancé***
Donnez la valeur de n:
Entrer n : 5
5!=120
***Algorithme terminé***

Généré par AlgoBox
```

FIGURE 9 – Résultat affiché dans l'exercice 9

1.10 Exercice 10

Ecrire un algorithme qui détermine le plus petit et le plus grand élément d'une liste de n nombres quelconques ainsi que leurs positions respectives.

Exercice 10

```
1 Algorithme EX10;
2 VARIABLES
3   i,n,max,min,pmax,pmin,nb:entier;
4 DEBUT
5   Ecrire("Donner le nombre de valeur n: ");
6   Lire(n);
7   Ecrire("Donner la valeur du premier element: ");
8   Lire(nb);
9   max <-- nb;
10  min <-- nb;
11  pmax <-- 1;
12  pmin <-- 1;
13  POUR i <-- 2 A n Faire
14    SI(nb>=max) Alors
15      Debut
16        max <-- nb;
17        pmax <-- i;
18      Fin;
19    Sinon Si(nb<=min) Alors
20      Debut
21        min <-- nb;
22        pmin <-- i;
23      Fin;
24    Fsi;
25  Fait;
26  Ecrire("La valeur maximale est:\n",max);
27  Ecrire("Sa position est:\n",pmax);
28  Ecrire("La valeur minimale est:\n",min);
29  Ecrire("Sa position est:\n",pmin);
30  FIN.
```

```
RÉSULTATS :
***Algorithme lancé***
Donnez la valeur de n:
Entrer n : 5
Entrer x : 1
Entrer x : 3
Entrer x : 2
Entrer x : 6
Entrer x : 4
La valeur max est:6
Sa position est: 4
La valeur min est: 2
Sa position est:3
***Algorithme terminé***

Généré par AlgoBox
```

FIGURE 10 – Résultat affiché dans l'exercice 10

1.11 Exercice 11

Ecrire un algorithme qui calcule la valeur de :

- a) $\cos(x) = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \dots$
b) $\sin^2(x) = 2 \times \frac{x^2}{2!} - 2^3 \times \frac{x^4}{4!} + 2^5 \times \frac{x^6}{6!} - \dots$

Exercice 11 Cos(x) : Méthode 1

```
1  Algorithme EX11_Cos;
2  VARIABLES
3  puiss,y,x:reel;
4  i,fact,n,sign:entier;
5  DEBUT
6  Ecrire("donnez les valeurs de x et n: ");
7  LIRE(x,n);
8  puiss <-- 1;
9  y <-- 1;
10 fact <-- 1;
11 sign <-- 1;
12 i <-- 0;
13 TANT_QUE (i<n) FAIRE
14   fact <-- fact*(i+1)*(i+2);
15   puiss <-- puiss*x*x;
16   sign <-- sign*(-1);
17   y <-- y+sign*puiss/fact;
18   i <-- i+2;
19 Fait;
20 Ecrire("la valeur de cos(x) vaut: ",y);
21 FIN.
```

Exercice 11 Cos(x) : Méthode 2

```
1  Algorithme EX11_Cos;
2  VARIABLES
3  puiss,y,x:reel;
4  i,fact,n,sign:entier;
5  DEBUT
6  Ecrire("donnez les valeurs de x et n: ");
7  LIRE(x,n);
8  puiss <-- 1;
9  y <-- 1;
10 fact <-- 1;
11 sign <-- 1;
12 Pour i <-- 1 à n FAIRE
13   fact <-- fact*(2*i)*(2*i-1);
14   puiss <-- puiss*x*x;
15   sign <-- sign*(-1);
16   y <-- y+sign*puiss/fact;
17 Fait;
18 Ecrire("la valeur de cos(x) vaut: ",y);
19 FIN.
```

RÉSULTATS :

```
***Algorithme lancé***
donnez les valeurs de x et n:
Entrer x : 2
Entrer n : 20
la valeur de cos(x) vaut: 1.0753391
***Algorithme terminé***
```

Généré par AlgoBox

FIGURE 11 – Résultat affiché dans l'exercice 11(Cos(x))

Exercice 11 Sin2(x) : Méthode 1

```
1  Algorithme EX11_Sin2;
2  VARIABLES
3  puiss1,puiss,y,x:reel;
4  i,fact,n,sign:entier;
5  DEBUT
6  Ecrire("donnez les valeurs de x et n: ");
7  LIRE(x,n);
8  puiss1 <-- 1;
9  puiss <-- 2;
10 y <-- 0;
11 fact <-- 1;
12 sign <-- 1;
13 i <--0;
14 TANT_QUE (i<n) FAIRE
15   fact <-- fact*(i+1)*(i+2);
16   puiss1 <-- puiss1*x*x;
17   y <-- y+sign*puiss/fact;
18   sign <-- sign*(-1);
19   puiss <-- puiss*2*2;
20   i <-- i+2;
21 Fait;
22 Ecrire("la valeur de cos(x) vaut: ",y);
23 FIN.
```

Exercice 11 Sin2(x) : Méthode 2

```
1  Algorithme EX11_Sin2;
2  VARIABLES
3  puiss1,puiss,y,x:reel;
4  i,fact,n,sign:entier;
5  DEBUT
6  Ecrire("donnez les valeurs de x et n: ");
7  LIRE(x,n);
8  puiss1 <-- 1;
9  puiss <-- 2;
10 y <-- 0;
11 fact <-- 1;
12 sign <-- 1;
13 Pour i <-- 1 à n FAIRE
14   fact <-- fact*(2*i)*(2*i-1);
15   puiss1 <-- puiss1*x*x;
16   y <-- y+sign*puiss/fact;
17   sign <-- sign*(-1);
18   puiss <-- puiss*2*2;
19 Fait;
20 Ecrire("la valeur de cos(x) vaut: ",y);
21 FIN.
```

RÉSULTATS :

```
***Algorithme lancé***
donnez les valeurs de x et n:
Entrer x : 1
Entrer n : 120
la valeur de sin2(x) vaut: 1
***Algorithme terminé***
```

Généré par AlgoBox

FIGURE 12 – Résultat affiché dans l'exercice 11($\sin^2(x)$)

1.12 Exercice 12

Soit une liste de nombres tous différents de zéro sauf le dernier. Ecrire l'algorithme qui calcule la moyenne des nombres positifs de cette liste et affiche un message d'erreur au cas où il en n'existerait pas.

Exercice 12

```
1 Algorithme EX12;
2 VARIABLES
3 nb,som,np: entier;
4 moy:reel;
5 DEBUT
6 np <-- 0;
7 som <-- 0; //Ceci est un commentaire
8 nb <-- 1; //N'importe quelle valeur different de zero
9 TANT_QUE (nb<>0) FAIRE
10  LIRE(nb);
11  SI (nb>0) ALORS
12    DEBUT
13      som <-- som+nb;
14      np <-- np+1; //Compteur de nombre positive
15    Fin;
16  Fsi;
17 Fait;
18 SI (np<>0) ALORS
19  DEBUT
20    moy <-- som/np;
21    Ecrire("la moyenne vaut:",moy);
22  Fin;
23 SINON Ecrire("Erreur aucun nombre positive");
24 FSI;
25 FIN.
```

RÉSULTATS :

```
***Algorithme lancé***
Entrer x : 1
Entrer x : -1
Erreur x<0
Entrer x : 2
Entrer x : 3
Entrer x : -4
Erreur x<0
Entrer x : 4
Entrer x : 0
Erreur x<0la moyenne vaut:2.5
***Algorithme terminé***
```

Généré par AlgoBox

FIGURE 13 – Résultat affiché dans l'exercice 12

1.13 Exercice 13

Soit une liste de nombres tous différents de zéro sauf le dernier. Ecrire l'algorithme qui indique la valeur et la position du premier élément négatif de cette liste. Ecrire un message d'erreur au cas où aucun élément négatif ne figure dans cette liste.

Exercice 13

```
1  Algorithme EX13;
2  VARIABLES
3  nb,pos,i,neg: entier;
4  DEBUT
5    nb <-- 1;
6    i <-- 0;
7    TANT_QUE (nb>0) FAIRE
13     LIRE(nb);
14     i <-- i+1;
15  Fait;
16  SI (nb=0) ALORS Ecrire("Erreur aucun nombre negative");
20     SINON DEBUT
22         pos <-- i;
23         neg <-- x;
24         Ecrire("la valeur négative est: ",neg);
26         Ecrire("Sa position est:",pos);
28  FSI;
29  FIN.
```

RÉSULTATS :

```
***Algorithme lancé***
Entrer x : 1
Entrer x : 2
Entrer x : 3
Entrer x : -1
la valeur négative est: -1
Sa position est:4
***Algorithme terminé***
```

Généré par AlgoBox

FIGURE 14 – Résultat affiché dans l'exercice 13

1.14 Exercice 14

Soit une liste de nombres tous différents de zéro sauf le dernier. Ecrire l'algorithme qui indique si une valeur K donnée figure dans cette liste.

Exercice 14

```
1  Algorithme EX14;
2  VARIABLES
3  nb,k:entier;
4  DEBUT
5    Ecrire("Doonez la valeur de k:");
6    LIRE(k);
7    nb <-- 1;
8    TANT_QUE (nb<>0) FAIRE
9      LIRE(nb);
10     SI (nb=k) ALORS
11       DEBUT
12         Ecrire("Oui la valeur k =",k);
13         Ecrire("appartient à cette liste");
14       Fin;
15     Fsi;
16   Fait;
17 FIN.
```

RÉSULTATS :

```
***Algorithme lancé***
Doonez la valeur de k:
Entrer k : 5
Entrer x : 2
Entrer x : -1
Entrer x : 3
Entrer x : 4
Entrer x : 5
Oui la valeur k =5 appartient à cette liste
Entrer x : 0
***Algorithme terminé***
```

Généré par AlgoBox

FIGURE 15 – Résultat affiché dans l'exercice 14