

## Solution Série 2 :

### Exercice 1:

1/ Non.

2/ Algorithme binaire;

Var: A, B, S, I: entier;

Début:

    lire(A, B);

    S ← A;

    I ← 1;

Tantque ( $I \leq B$ ) faire

    S ← S + I;

    I ← I + 1;

Fois;

    Ecrire(S);

Fin.

3) On a utilisé la boucle Tantque car on n'a pas le nombre de répétition (n'est pas connu salement la condition  $I \leq B$ ).

### Exercice 2 :

1/ Développement:

Instruction	N	nbr	S	I
lire(N)	4			
lire(nbr)	4	10		
lire(nbr)	4	15		
lire(nbr)	4	22		

lire(nbr)	4	8	4	
Ecrire(S)				
$S \leftarrow 0$	4	8	8	1
$S \leftarrow S + nbr$	4	8	8	2
$S \leftarrow 0$	4			
$S \leftarrow S + nbr$		8	8	3
$S \leftarrow 0$	4			
$S \leftarrow S + nbr$		8	8	4
Ecrire(S)	4	8	8	

2/ le raisonnement erroné est :

1er :  $S \leftarrow 0$  avant la boucle

2eme : lire(nbr) il faut le contenir au cours de la 2eme boucle qui calcule la somme

la corrigé de l'algorithme:

$S \leftarrow 0;$

Pour  $I \leftarrow 1$  à N faire

    lire(nbr);

$S \leftarrow S + nbr;$

    Ecrire( $\rightarrow S$ );

Fpour ..

### Exercice 3 :

Algorithme Somme ;

Var: n, s, i : entier

Début:

    Ecrire("donner un nombre!");

brie (A);

Si  $N > 0$  alors

$S \leftarrow 0;$

Pour  $i \leftarrow 1$  à  $N$  faire

$S \leftarrow S + i;$

Fpour;

Ecrire ("la somme est", S);

Fin. Sinon

Ecrire ("Le nombre est négatif");

Fin;

Fin.

Exercice 4:

Algorithme Puissance;

Var: a, p, n, i : entier;

Début:

Ecrire ("donner un nombre");

brie (a);

Ecrire ("donner l'apuissance");

brie (n);

Si  $n = 0$  alors

$p \leftarrow 1;$

Sinon

$p \leftarrow 1;$

Pour  $i \leftarrow 1$  à  $n$  faire

$p \leftarrow p * a;$

Fpour

FFin

Ecrire ("la puissance est", p);

Fin.

Exercice 5:

Algorithme Exo5;

Var: A, B, R, I : entier;

Début:

Repéter

brie (A, B)

jusqu'à ( $A = 0$  et  $B > 0$ )

$R \leftarrow A;$

Tantque ( $R \geq B$ ) faire

$R \leftarrow R - B;$

FTq,

Ecrire ("le reste de la division  
est", R);

Fin.

Exercice 6:

Algorithme Exo6;

Var: A, R, I;

Début:

brie (A);

$R \leftarrow A$

Tantque ( $R >= 0$ ) faire

$R \leftarrow R - 2$

FTq,

Si ( $R = 0$ ) alors

Ecrire ("A'' est divisible par 2'');

Sinon

Ecrire ("A'' n'est pas divisible par 2");

Fini

Fin.

Eercice 7:

Algorithme MinSuit;

Var: n, A, i : entier;

Début:

Ecrire ("donner un nombre");

lire (n);

Ecrire ("donner un autre");

lire (A);

Min  $\leftarrow$  A;

Pour i  $\leftarrow$  1 à N faire

    lire (A);

    Si (Min > A) alors

        Min  $\leftarrow$  A;

    Fsi

Fpour

Ecrire ("le minimum est", Min);

Fin.

Eercice 8:

Algorithme Premier;

Var: n, i, r : entier;

Début:

lire (n);

r  $\leftarrow$  0;

Pour i  $\leftarrow$  2 à n faire

    Si ( $n \bmod i = 0$ ) alors

        r  $\leftarrow r + 1$ ;

Fsi

Fpour

Si (r = 2) alors

Ecrire ("n est un nombre premier");

Sinon

Ecrire ("n n'est pas premier");

Fsi

Fin.

Eercice 9:

Algorithme RICD;

Var: a, b : entier;

Début:

lire (a, b);

Tantque ( $a * b <= 0$ ) faire

    Si ( $a \geq b$ ) alors

        a  $\leftarrow a - b$ ;

    Sinon

        b  $\leftarrow b - a$ ;

    Fsi

FTq.

Si ( $a = 0$ ) alors

Ecrire ("le paquet", b);

Sinon

Ecrire ("le paquet", a);

Fsi

Fin

Exercice 10 :

Algorithme Suite;

Var;  $u_n, m_i, n, i$ ; réel;

Début:

lire (n);

$m_i \leftarrow 2023$ ;

Pour  $i \leftarrow 2$  à  $n$  faire

$u_n \leftarrow 1000/m_i + u_{i-1}/1000$ ;

$m_i \leftarrow m_{i-1}$ ;

Fpour

Ecrire ( $u_n$ );

Fin.

Exercice 11 :

Algorithme Some ;

Var: S, F, K, i, n

Début:

lire (n);

$S \leftarrow 1; F \leftarrow 1; K \leftarrow 1$ ;

Pour  $i \leftarrow 1$  à  $n$  faire

$F \leftarrow F * i$ ;

$K \leftarrow -K$

$S \leftarrow S + K / (2 * i + 1) * F$ ;

Fpour

Ecrire (S);

Exercice 12 :

Algorithme clé;

Var: N, r : entier;

Début:

$r \leftarrow 0$ ;

Ecrire ("donner un nombre");

lire (N);

Tant que ( $N > 0$ ) faire

$r \leftarrow r * 10$ ;

$r \leftarrow r + N \bmod 10$ ;

$N \leftarrow N / 10$ ;

Ffq

Ecrire (R);

Fin.