ESATIC ECOLE SUPÉRIEURE AFRICAINE DES TIC	Année scolaire 2014 - 2015 PROF : KONAN HYACINTHE LICENCE 1 SRIT

CHAPITRE 1 :NOTION DE BASE
TRAVAUX DIRIGES

EXERCICE 1. Chercher l'erreur

Relevez les erreurs dans l'algorithme suivant en les expliquant.

```
ALGORITHME AlgoACorriger

        CONSTANTE
                PI = 3.14
        VARIABLES
                m : ENTIER
                n, p, q : REEL
                c, d, 1x : CARACTERE
                b1, b2 : BOOLEEN
DEBUT
        m ← 7
        p ← n + p
        c ← 'u'
        x ← 2.5
        b1 ← c != 'r'
        b2 ← (m == 7) OU b1
        n ← m * PI
        m * 3 ← m + 5
        p = 7.0
        PI ← 3.14159
        q ← 3m
FIN
```

EXERCICE 2. * Simulation instructions élémentaires

Réaliser deux simulations de l'algorithme suivant avec des données différentes :

```
ALGORITHME Corde
/* Déclarations */
VARIABLE

        perimetre : ENTIER /* périmètre de la poulie */
        lgCorde : ENTIER /* longueur de la corde */
        nbTours : ENTIER /* nombre de tours */
        lgReste : ENTIER */ longueur restante */
DEBUT
        /* entrée des données */
1       ECRIRE ("Quel est en cm le périmètre de la poulie (un entier) ")
2       LIRE (perimetre)
3       ECRIRE ("Quel est en cm la longueur de la corde (un entier) ")
4       LIRE (lgCorde)
        /* calculs */
5       nbTour ← lgCorde DIV perimetre
6       lgReste ← lgCorde MOD perimetre
        /* affichage des résultats */
7       ECRIRE ("on peut faire ", nbTour, "tours")
8       ECRIRE ("il reste ", lgreste, " cm non enroulés")
FIN
```

Attention ! : les numéros de ligne ne font pas partie de l'algorithme. Ils ne sont présents que pour faciliter la simulation.

ligne	perimetre	lgCorde	nbTours	lg_reste	écran
1					
2					
3					
4					
5					
6					
7					
8					
1					
2					
3					
4					
5					
6					
7					
8					

EXERCICE 3. * Simulation instructions élémentaires

Ecrire les déclarations et simuler les algorithmes suivants (numéroter les lignes et construire les tableaux)

```
ALGORITHME : Calcul_Prix1

DEBUT
        prix ← 15
        taux ←  10.5
        remise ←  (prix * taux)
        remise ←  remise / 100
        prix ←  prix - remise
        ECRIRE ( prix )
FIN
```

```
ALGORITHME : Calcul_Prix1

DEBUT
        a ← 10
        b ← 70
        q1 ← (a + b) / 5
        q2 ← (a + b) DIV 5
        r2 ← (a + b) MOD 5
        ECRIRE (q1, q2, r2)
        q1 ← (a + b) / 3
        q2 ← (a + b) DIV 3
        r2 ← (a + b) MOD 3
        ECRIRE (q1, q2, r2)
FIN
```

EXERCICE 4. * Franc CFA

Ecrire un algorithme qui demande une somme exprimée en EURO puis calcule et affiche son équivalent en francs CFA.

EXERCICE 5. ** Echange et permutation circulaire

1) Ecrire la séquence d'instructions qui échange le contenu des deux variables. Le but de cet exercice est d'échanger le contenu de deux variables et non pas de les intervertir à l'affichage. L'affichage des variables avant et après l'échange sert seulement à vérifier que l'échange a bien eu lieu.

```
ALGORITHME EchangeV1
/* Déclarations */
VARIABLE
        var1 : ENTIER
        var2 : ENTIER
DEBUT
        LIRE (var1, var2)
        ECRIRE (var1, var2)
        /* les valeurs des variables apparaissent dans l'ordre de leur saisie */
…
        ECRIRE (var1, var2)
        /* ici elles apparaissent dans l'ordre inverse de leur saisie */
FIN
```

2) Ecrire la séquence d'instructions qui effectue une permutation circulaire à droite du contenu des trois variables. Si le premier affichage donne " 3,5 6,2 10,4 ", le deuxième devra donner " 10,4 3,5 6,2 "

```
ALGORITHME EchangeV2
/* Déclarations */
VARIABLE
        var1 : REEL
        var2 : REEL
        var3 : REEL
DEBUT
        LIRE (var1, var2, var3)
        ECRIRE (var1, var2, var3)
       /* les valeurs des variables apparaissent dans l'ordre de leur saisie */
…
        ECRIRE (var1, var2, var3)
       /*ici elles apparaissent dans l'ordre inverse de leur saisie */
FIN
```

EXERCICE 6. *** Position d'un jeton sur un damier

Sur un damier rectangulaire de n lignes et p colonnes (n et p connus) on dispose, ligne après ligne, (n * p) jetons numérotés de 1 à n * p. Ecrire un algorithme qui demande la valeur du jeton et renvoie le numéro de ligne et le numéro de colonne de la case où il se

	1	2	**3**	4	5	6
1	1	2	3	4	5	6
2	7	8	9	10	11	12
3	13	14	15	16	17	18
4	19	20	**21**	22	23	24
5	25	26	27	28	29	30

FIN

ESATIC ECOLE SUPÉRIEURE AFRICAINE DES TIC	Année scolaire 2014 - 2015 PROF : KONAN HYACINTHE LICENCE 1 SRIT

CHAPITRE 1 :NOTION DE BASE
TRAVAUX DIRIGES

CORRECTION

EXERCICE 1. Chercher l'erreur

Algorithme AlgoACorriger
Constantes
PI = 3.14
Variables
m : entier
n : réel
p : réel
q : réel
c : caractère
d : caractère
b1 : booléen
b2 : booléen
1x : caractère
/* 1x n'est pas un identificateur correct : ne doit pas commencer par un chiffre */

Début
m ← 7
p ← n + p ***/* n n'est pas initialisé */***
c ← 'u'
x ← 2.5 ***/* x n'est pas déclaré */***
b1 ← c != 'r'
b2 ← (m == 7) OU b1
n ← m * PI
m * 3 ← m + 5
/* une variable doit être en partie gauche d'une affectation */
p = 7.0
PI ← 3.14159
/* pas d'affectation sur PI qui est une constante */
q ← 3m

/* 3m est un identificateur ni valable, ni connu. Si cela doit correspondre à un produit, il faut utiliser l'opérateur * : 3 * m */

Fin

EXERCICE 2. * Simulation instructions élémentaires

ligne	perimetre	lgCorde	nbTours	lg_reste	écran
1					Quel est en cm le périmètre de la poulie (un entier) ?
2	*10*				
3					Quel est en cm la longueur de la corde(un entier) ? (*)
4		*94*			
5			*9*		
6				*4*	
7					on peut faire 9 tours
8					il reste 4 cm non enroulés
					Simulation avec d'autres valeurs
1					Quel est en cm le périmètre de la poulie (un entier) ?
2	*13*				
3					Quel est en cm la longueur de la corde(un entier) ? (*)
4		*143*			
5			*11*		
6				*0*	
7					on peut faire 11 tours
8					il reste 0 cm non enroulés

EXERCICE 3. * Simulation instructions élémentaires

	prix	**taux**	**remise**	**écran**
1	15			
2		10.5		
3			157.5	
4			1.575	
5	13.425			
6				13.425

	a	**b**	**q1**	**q2**	**r2**	**écran**
1	10					
2		70				
3			16			
4				16		
5					0	
6						16 16 0
7			26.666			
8				26		
9					2	
10						26.666 26 2

EXERCICE 4. * Franc CFA

```
Algorithme Euro
    Constantes
    UN_EURO = 6.55957 /* valeur de l'euro en francs */
    Variables
    sEuro : réel /* somme en euros */
    sFranc : réel /* somme en francs */
Début
    écrire("somme en euros ") /* entrées des données */
    lire(sEuro)
    sFranc ← sEuro * UN_EURO /* traitement */
    écrire ("valeur en francs : ", sFranc) /* sortie des résultats*/
Fin
```

EXERCICE 5. ** Echange et permutation circulaire

1) il faut déclarer une variable intermédiaire, par exemple, aux de type entier.
aux ← var1
var1 ← var2
var2 ← aux

2) il faut déclarer une variable intermédiaire, par exemple, aux de type réel.
aux ← var1
var1 ← var3
var3 ← var2
var2 ← aux
D'autres séquences sont possibles.

EXERCICE 6. *** Position d'un jeton sur un damier

```
Algorithme Jeton
    Variables
        valeur : entier /* valeur du jeton */
        n : entier /* nombre de lignes du damier */
        p : entier /* nombre de colonnes du damier */
        lig : entier /* numéro de lignes du jeton */
        col : entier /* numéro de colonnes du jeton */
        /* Instructions */
Début
    écrire("nombre de lignes") /* entrées des données */
    lire(n)
    écrire("nombre de colonnes")
    lire(p)
    écrire("valeur du jeton")
    lire(valeur) /* supposée appartenir au damier */
    col ← 1 + (valeur – 1) mod p /* traitement */
    lig ← 1 + (valeur – 1) div p
    écrire(lig, col) /* sortie des résultats */
Fin
```

Autre solution : on peut remplacer les deux affectations du traitement par l'instruction conditionnelle :

```
si valeur mod P = 0 alors /* on est sur la dernière colonne */
    col ← p
    lig ← valeur div p
sinon
    col ← valeur mod p
    lig ← 1 + valeur div p
finsi
```

FIN