

### Contrôle de la session normale

#### Exercice 1 (8 Pts.)

Soit l'algorithme suivant:

Algorithme Calcul

Déclarations

$N, p, i, S, X, a$  entier

Début

Écrire(N, a)

$S \leftarrow (N \times 0) \text{ ou } (a \times 0)$  afin

Écrire(S, format(' '))

Si

$N < 0$

$p \leftarrow 1$

$X \leftarrow 1$

Écrire ( $X$ , format(' ')) afin

$S \leftarrow p \times X$

$X \leftarrow p \times X$

$N \leftarrow N - 1$

Fin

Écrire(S)

Fin

Fin

1. Dérouler manuellement l'algorithme pour:

a)  $N=4$  et  $a=3$

b)  $N=3$  et  $a=2$

2. Que calcule cet algorithme ?

3. Transformer l'algorithme ci-dessus en un programme Fortran.

#### Exercice 2 (4 Pts)

Evaluer les expressions suivantes:

a.  $3 \times 4 + (2 \times 15 - 6 \times 3) - 31 - 7 \times 6 / 241$

b.  $(5 \times p + 3 \times q - 1) \times r + (10 - 2 \times m) \times n - 1 \times a \times b + (2 \times e) \times f \times (3 \times g)$

**Résoudre deux (02) parmi les trois exercices qui suivent.**

#### Exercice 3 (4 Pts)

Soit la suite  $(U_n)$  définie par:

$$U_0 = 3$$

$$U_n = 2U_{n-1} - 2 \text{ quand } n > 0$$

Ecrire un algorithme qui affiche les termes de  $(U_n)$  dont la valeur est inférieure à 200.

#### Exercice 4 (4 Pts)

Soit la matrice réelle M constituée de 50 lignes et 70 colonnes.

Ecrire en Fortran une fonction qui calcule la moyenne des éléments appartenant aux lignes paires (les lignes 2,4,6, ...,16,18,50).

#### Exercice 5 (4 Pts)

Ecrire en Fortran un programme qui utilise une fonction fortran calculant le discriminant "Delta" pour vérifier si chacune parmi 10 équations du deuxième degré  $Ax^2 + Bx + C = 0$  possède ou non des racines réelles.

(Les coefficients A, B et C sont donnés sous forme de vecteurs).

Bon courage !

**Corrigé type du contrôle de la session normale**

**Exercice 1 (8 Pts.)**

1. 1.5 Pts. X 2

i	p	F	S	N	a	Ecran
1	1	1	0	4	3	2127
2	3	1	3			
3	9	2	21			
4	27	6	183			
5	81	24	2127			
1	1	1	0	3	-2	-42
2	-2	1	-2			
3	4	2	6			
4	-8	6	-42			

2. L'algorithme calcule  $\sum_{i=1}^N a \cdot i!$  (1 Pts.)

3. (4 Pts. - 0.25 Pts. par erreur)

```

program Calcul
integer p, F, S, N, a
read(*, *) N, a
if (N <= 0) or (a <= 0) then
    write(*, *) 'Erreur'
else
    S := 0
    F := 1
    p := 1
    do i = 1, N
        p := p * a
        F := F * i
        S := S + p * F
    enddo
    write(*, *) S
endif
end
    
```

**Exercice 3 (4 Pts.)**

Algorithme suite

Déclarations

U: Entier

Début

U ← 3

TantQue U < 200 faire

    Ecrire(U)

    U ← 2\*U - 2

FinTantQue

Fin

Valeur initiale (0.5 Pts.)

Boucle (1.5 Pts.)

Ecriture et calcul du terme suivant (2 Pts.)

**Exercice 4 (4 Pts.)**

```

real fonction Moyenne(M)
real M(50, *)
real S
integer i, j
S := 0
do i = 1, 50, 2
    do j = 1, 70
        S := S + M(i, j)
    enddo
enddo
Moyenne := S / (25 * 70)
return
end
    
```

Entête (1 Pts.)

Parcours de la matrice (1 Pts.)

Calcul de la somme (1 Pts.)

Affectation du résultat au nom de la fonction (0.5 Pts.)

Return (0.5 Pts.)

**Exercice 5 (4 Pts.)**

```

program Equations
real A(10), B(10), C(10)
integer
Delta(a, b, c) = b**2 - 4*a*c
do i = 1, 10
    read(*, *) A(i), B(i), C(i)
enddo
do i = 1, 10
    if Delta(A(i), B(i), C(i)) <= 0 then
        write(*, *) 'L''équation', i, 'possède
des racines réelles'
    else
        write(*, *) 'L''équation', i, 'ne
possède pas des racines réelles'
    endif
enddo
end
    
```

Lecture (0.75 Pts.)

Déclaration de Delta (1 Pts.)

Test pour chaque équation (1.5 Pts.)

Affichage (0.75 Pts.)

Exercise 2. (4 Pts)

