

## TD/TP2: Les Structures Conditionnelles

### 1. INTRODUCTION

En programmation, on est souvent confronté à des situations où on a besoin de choisir entre deux ou plusieurs alternatives pour cela on utilise une structure conditionnelle

### 2. TYPES D'INSTRUCTIONS CONDITIONNELLES

Une Structure conditionnelle permet à l'algorithme de modifier son traitement en fonction d'une condition. Elle permet d'exécuter ou non un bloc d'instructions selon le résultat d'un test. Les structures conditionnelles peuvent être classifiées comme suit :

#### 2.1 Structure Conditionnelle Simple (à un seul choix)

##### Syntaxe

**Si ( Cond) Alors**

<Séquence d'instructions 1>

**FinSi**

#### 2.2 Structure Conditionnelle Alternative (à deux choix)

##### Syntaxe

**Si (Cond) Alors**

<Séquence d'instructions 1>

**Sinon**

<Séquence d'instructions 2>

**FinSi**

#### 2.3 Structure Conditionnelle Imbriquée (à choix multiples)

##### Syntaxe

**Si (Cond1) Alors**

<Séquence d'instructions 1>

**Sinon**

**Si (Cond2) Alors**

<Séquence d'instructions 2>

**Sinon**

<Séquence d'instructions 3>

**FinSi**

**FinSi**

### Exercice 1

Ecrire un algorithme qui permet de calculer l'inverse d'un entier s'il est non nul.

### Exercice 2

Ecrire un algorithme qui lit un entier et affiche ensuite s'il est pair ou impair.

### Exercice 3

Ecrire un Algorithme qui lit le potentiel hydrogène d'une solution puis affiche sa nature selon les cas suivants :

-Ph inférieur à 7 afficher que la solution est acide

-Ph supérieur à 7 afficher que la solution est basique

-Ph égal à 7 afficher que la solution est neutre

### Exercice 4

Ecrire un algorithme qui demande deux nombres à l'utilisateur et l'informe ensuite si le produit est négatif ou positif (on inclut le traitement du cas où le produit peut être nul).

### Exercice 5

Etant donné **X** un réel, écrire un algorithme qui permet de calculer **Y** selon les cas suivants

$$Y = \begin{cases} 2X^3 + 3 & \text{si } X < 0 \\ 3 - 4X & \text{si } 0 \leq X \leq 2 \\ 3 - 2X^2 & \text{si } X > 2 \end{cases}$$

### Exercice 6

Ecrire un algorithme permettant de résoudre dans **R** une équation du second degré de la forme **ax<sup>2</sup>+bx+c = 0**

## TD/TP3: Les Structures Itératives

### 1.Introduction

La structure itérative (boucle) répète l'exécution d'une ou plusieurs instructions.

<b>2.La boucle Tant que ...Faire</b> <u>Structure :</u> <b>Tant que</b> (cond vraie) <b>faire</b> <Séquence d'instructions > <b>FinTQ</b>	<b>3.La boucle Répéter Jusqu'à</b> <u>Structure :</u> <b>Répéter</b> <Séquence d'instructions> <b>Jusqu'à</b> (COND)	<b>4.La boucle Pour ...Faire</b> <u>Structure :</u> <b>Pour</b> compt ← v initiale à v finale <b>Faire</b> <Séquence d'instructions> <b>FinPour</b>
---	--	---

### Exercice 1

Tracer l'exécution des morceaux des algorithmes suivants

```

I←0
S←0
Tantque (I <=5) faire
I←I+1
S← S+ I+1
Ecrire (S)
Ecrire (I)
FINTQ
    
```

```

I←0
S←1
Répéter
I←I+2
S← S* I
Ecrire (I)
Ecrire (S)
Jusqu'à (I >7)
    
```

```

I←1
Tantque (I <=5) faire
Ecrire (i)
Finque
    
```

```

Pour I ← de 6 à 1
Ecrire (I)
Finpour
    
```

```

S←0
Pour I ← de 1 à 6 faire
S← S+ I
Ecrire (I)
Ecrire (S)
finpour
    
```

### Exercice2

Ecrire un algorithme qui lit un entier positif n puis calcule et affiche son factoriel selon la formule  $n! = 1 * 2 * n$ . (via la boucle Pour, Tant que, Répéter)

### Exercice 3:

```

Algorithme somme
Var i,SP,SI,N : Entier
Début
Ecrire('donner la valeur de N')
Lire(N)
SI ← 0
SP ← 0
Pour i←1 à N faire
Si (i mod 2=0) alors
SP ←SP+i
Ecrire(SP)
Sinon
SI←SI+i
Ecrire(SI)
Finsi
Finpour
Fin
    
```

### Exercice 4

1. Ecrire un algorithme permettant de calculer l'expression suivantes via la boucle tant que

$$S = 1 - \frac{1}{4} + \frac{1}{9} - \frac{1}{16} + \dots \pm \frac{1}{n^2}$$

2. Dérouler l'algorithme pour N=4 en donnant le contenu de chaque variable après l'exécution

### Exercice 5

1. Etant donné X un réel et n un entier, écrire le programme qui permet de calculer la somme suivante via la boucle pour:

$$s = 1 + \frac{x^2}{2} + \frac{x^4}{2^2} + \frac{x^6}{2^3} \dots + \frac{x^{2n}}{2^n}$$

2. Dérouler l'algorithme pour N=3 en donnant le contenu de chaque variable après l'exécution

-Dérouler l'algorithme pour N=10

-Que calcule cet algorithme.