

1) Structures de contrôles

1.1) Structures de contrôle conditionnelles

Ces structures sont utilisées pour décider de l'exécution d'un bloc d'instruction : est-ce que ce bloc est exécuté ou non. Ou bien pour choisir entre l'exécution de deux blocs différents.

Nous avons deux types de structures conditionnelles :

1.1.1) Test alternatif simple

Un test simple contient un seul bloc d'instructions. Selon une condition (expression logique), on décide est-ce que le bloc d'instructions est exécuté ou non. Si la condition est vraie, on exécute le bloc, sinon on ne l'exécute pas. La syntaxe d'un test alternatif simple est comme suit :

<pre>si <Condition> alors <Bloc d'instruction(s)> finsi;</pre>	<pre>if <condition> then begin <bloc d'instruction (s)>; end;</pre>
-----------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------

1.1.2) Test alternatif double

Un test double contient deux blocs d'instructions : on est amené à décider entre le premier bloc ou le second. Cette décision est réalisée selon une condition (expression logique ou booléenne) qui peut être vraie ou fausse. Si la condition est vraie on exécute le premier bloc, sinon on exécute le second.

La syntaxe d'un test alternatif simple est :

<pre>si <Condition> alors <Bloc d'instruction(s) 1> sinon <Bloc d'instruction(s) 2> finsi;</pre>	<pre>if <condition> then begin <bloc d'instruction (s) 1>; end else begin <bloc d'instruction (s) 2>; end;</pre>
--------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Exercice 1 :

Soit l'algorithme suivant :

Algorithme test_alternative

Variables x : réel

Début

Lire(x)

Si (x mod 2 = 0) Alors

X ← X+1

Finsi

Ecrire(x)

Fin.

Questions

- 1) Dérouler l'algorithme pour x = 4 et x = 5.
- 2) Traduire l'algorithme en programme PASCAL.

Exercice 2 :

Ecrire un programme Pascal permettant d'afficher le coût d'impression d'un nombre de page np. Sachant que :

Tarif = 5 DA par page Si np < 10 ;

Tarif = 4 DA par page Si 10 ≤ np < 50 ;

Tarif = 3 DA par page Si np ≥ 50.