

Structures Conditionnelle (toujours)

Variables Booléennes

Nous avons vu qu'il existent des variables qui ont comme type : **Booléen**. L'utilisation des ces variables peut nous permettre de clarifier le code. En Effet, si nous reprenons l'algorithme de l'exercice précédent :

```
Variable ent1 : Entier
Variable ent2 : Entier
Début
    Afficher ( "Entrez première valeur : " )
    Saisir ( ent1 )
    Afficher ( "Entrez deuxième valeur : " )
    Saisir ( ent2 )
    Si ( ent1 = 0 Ou ent2 = 0 ) Alors
        Afficher ( "Le produit est nul" )
    Sinon
        Si ( ( ent1 > 0 Et ent2 > 0 ) Ou ( ent1 < 0 Et ent2 < 0 ) ) Alors
            Afficher ( "Positif" )
        Sinon
            Afficher ( "Négatif" )
        FinSi
    FinSi
Fin
```

Nous avons une condition qui peut paraître complexe, mais qui grâce à l'utilisation de deux variables booléennes, sera plus claire. Nous allons pour cela utiliser :

- une variable sontTousPositifs qui sera affectée par cette comparaison : $\text{ent1} > 0$ ET $\text{ent2} > 0$
- une variable sontTousNegatifs qui sera affectée par cette comparaison : $\text{ent1} < 0$ ET $\text{ent2} < 0$

Nous pourrons ensuite utiliser ces variables dans la structure conditionnelle précédente. Ce qui va donner l'algorithme suivant :

```
Variable ent1 : Entier
Variable ent2 : Entier
Variable sontTousPositifs : Booléen
Variable sontTousNegatifs : Booléen
Début
    Afficher ( "Entrez première valeur : " )
    Saisir ( ent1 )
    Afficher ( "Entrez deuxième valeur : " )
    Saisir ( ent2 )
    sontTousPositifs <- ent1 > 0 Et ent2 > 0
    sontTousNegatifs <- ent1 < 0 Et ent2 < 0
    Si ( ent1 = 0 Ou ent2 = 0 ) Alors
        Afficher ( "Le produit est nul" )
    Sinon
        Si ( sontTousPositifs Ou sontTousNegatifs ) Alors
            Afficher ( "Positif" )
        Sinon
            Afficher ( "Négatif" )
        FinSi
    FinSi
Fin
```