

CHAP.V : INSTRUCTIONS DE CONTRÔLE (Tests & Boucles) :

V.1. Opérateurs De Comparaison Et Opérateurs Logiques :

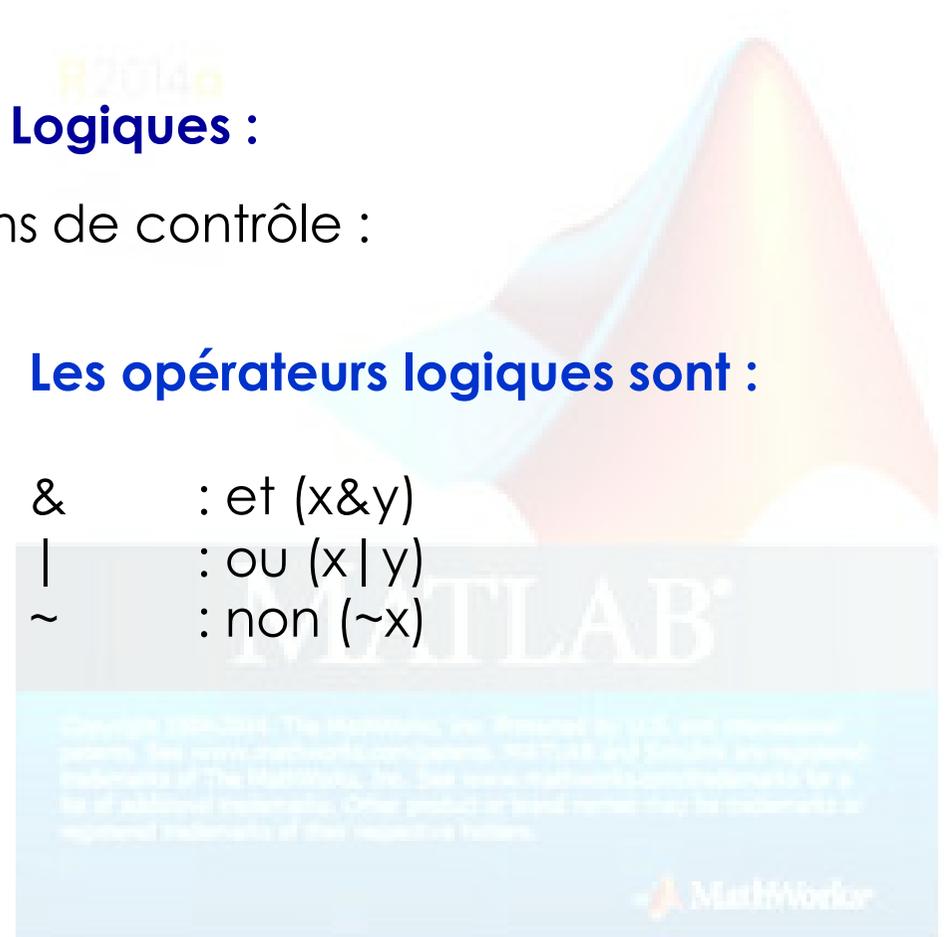
Ils sont utilisés essentiellement dans les instructions de contrôle :

Opérateurs de comparaison :

`==` : égal à ($x==y$)
`>` : strictement plus grand que ($x>y$)
`<` : strictement plus petit que ($x<y$)
`>=` : plus grand ou égal à ($x>=y$)
`<=` : plus petit ou égal à ($x<=y$)
`~=` : différent de ($x~=y$)

Les opérateurs logiques sont :

`&` : et ($x&y$)
`|` : ou ($x|y$)
`~` : non ($\sim x$)



CHAP.V

V.2. l'instruction conditionnée IF :

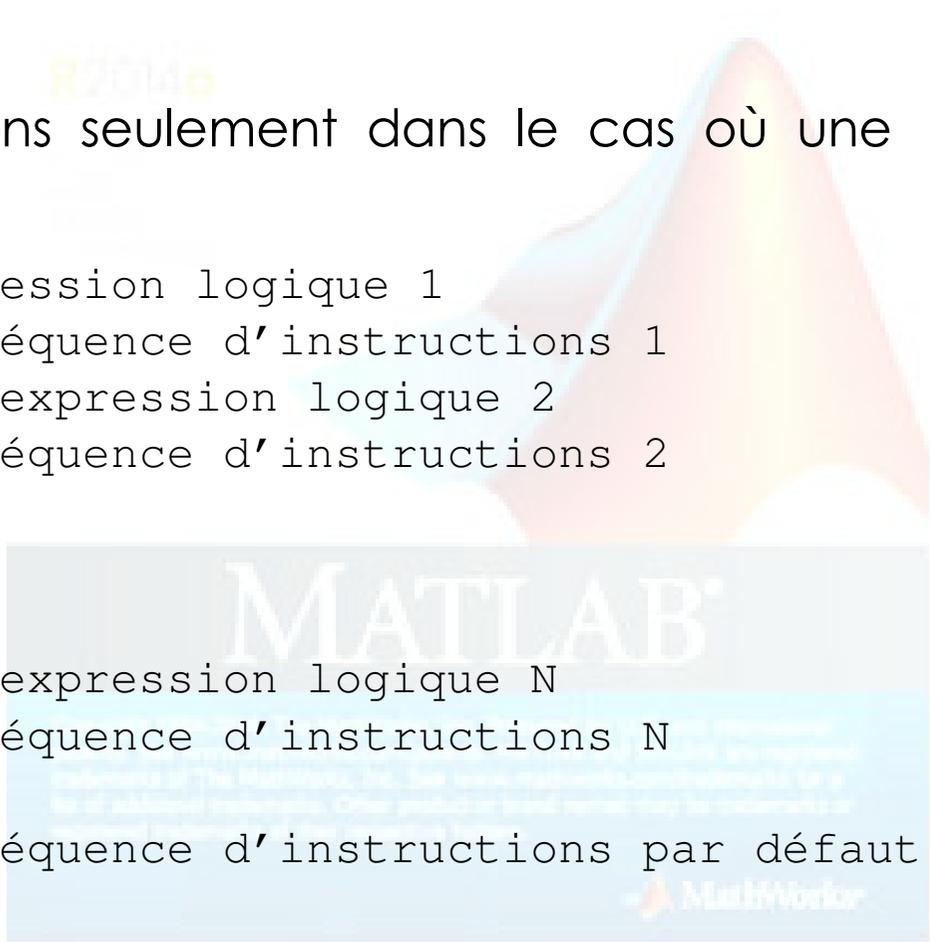
Permet d'exécuter une séquence d'instructions seulement dans le cas où une condition donnée est vérifiée au préalable.

Syntaxe :

```
if expression logique
    séquence d'instructions
end
```

```
if expression logique
    séquence d'instructions 1
else
    séquence d'instructions 2
end
```

```
if expression logique 1
    séquence d'instructions 1
elseif expression logique 2
    séquence d'instructions 2
.
.
.
elseif expression logique N
    séquence d'instructions N
else
    séquence d'instructions par défaut
end
```



CHAP.V

APPLICATION 1 (Tests) :

Soient **a** et **b** deux réels.
Calculer **y**, sachant que pour :

$$a = b \implies y = a^2$$

$$a > b \implies y = a.b^2$$

$$a < b \implies y = a.b$$

AN : $a = b = 3$; $a = 5$ et $b = 4$; $a = 2$ et $b = 3$



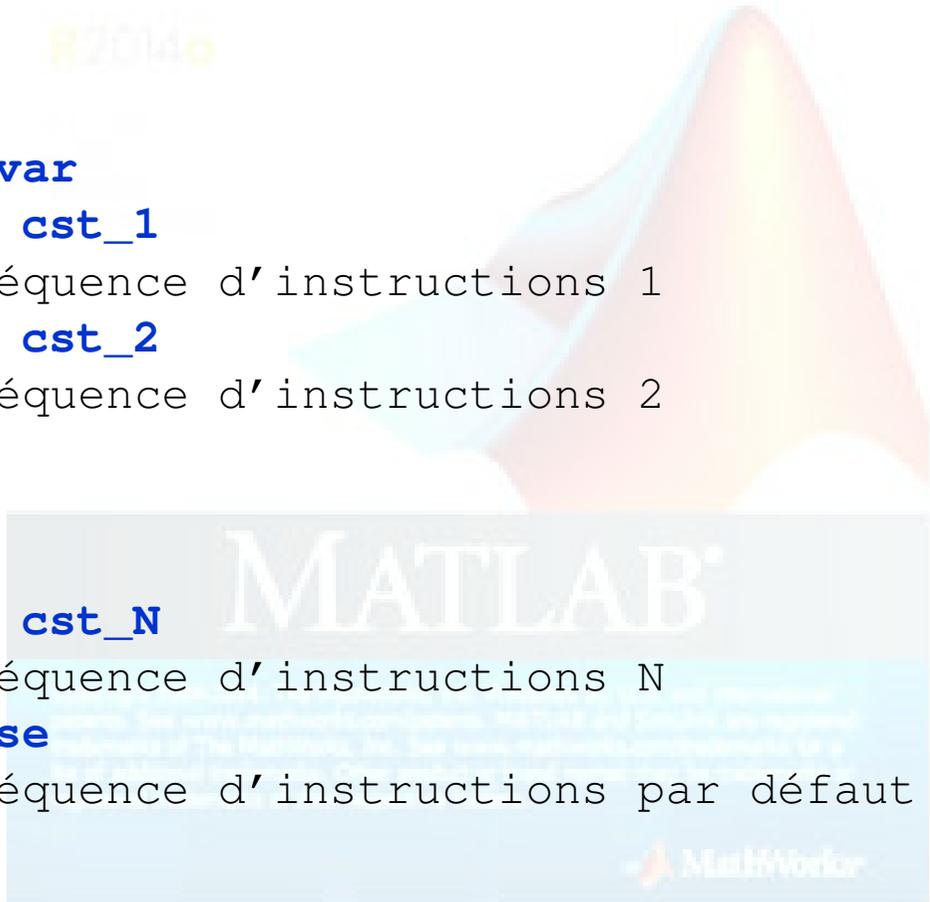
CHAP.V

V.3. Choix ventilé, l'instruction switch:

Permet l'utilisation d'une séquence d'instructions conditionnées pour effectuer un choix en cascade.

Syntaxe :

```
switch var
  case cst_1
    séquence d'instructions 1
  case cst_2
    séquence d'instructions 2
  .
  .
  .
  case cst_N
    séquence d'instructions N
otherwise
    séquence d'instructions par défaut
end
```

The background of the code block features a large, semi-transparent MATLAB logo. The logo consists of a stylized mountain peak with a color gradient from blue at the base to yellow at the top. Below the mountain, the word "MATLAB" is written in a large, white, serif font. At the bottom right of the logo area, the MathWorks logo is visible, which includes a stylized 'M' icon and the text "MathWorks".

CHAP.V

APPLICATION 2 (switch) :

Soient **a** et **b** deux réels.
Calculer **y**, selon le cas de **z** :

1^{er} Cas : $z = 16 \implies y = a + b$

2^{ème} Cas : $z = 17 \implies y = a + 2b$

3^{ème} Cas : $z = 18 \implies y = a + 3b$

Afficher « Aucun cas n'est vérifié » si **z** est différent des cas cités.

AN : $a = 3$ et $b = 4$



CHAP.V

V.4. Boucle FOR (pour ... faire) :

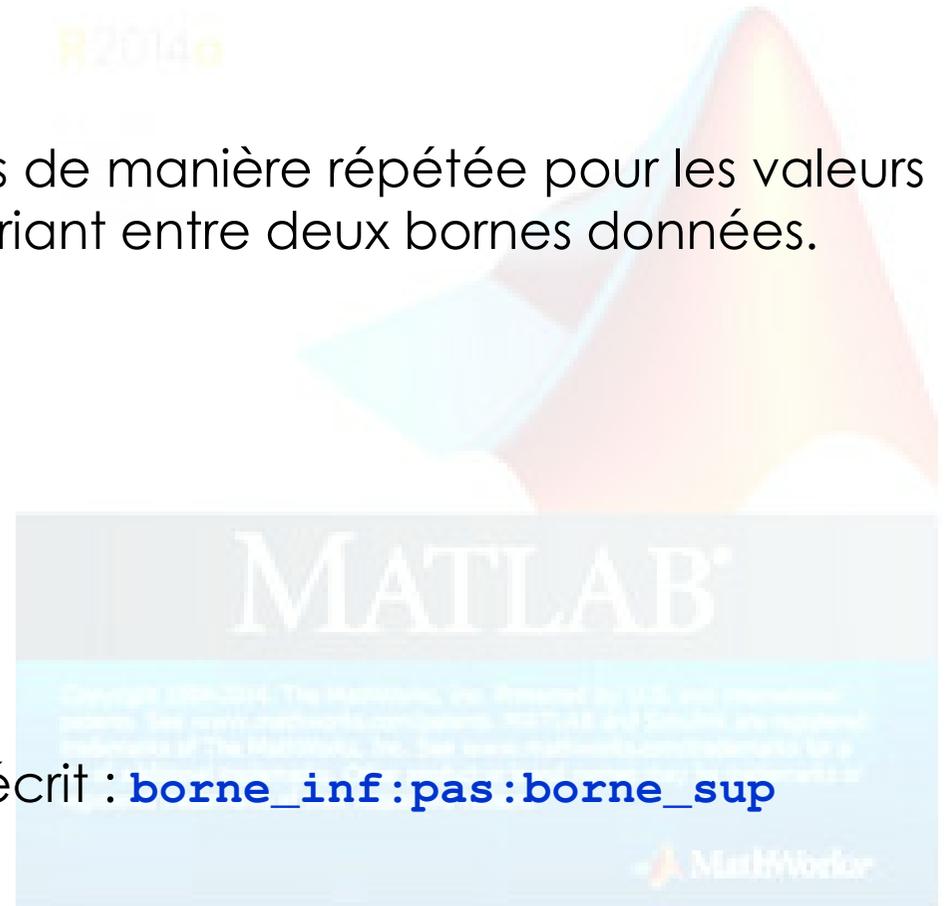
Permet d'exécuter une séquence d'instructions de manière répétée pour les valeurs d'un indice, incrémenté à chaque itération, variant entre deux bornes données.

Syntaxe :

```
for indice = borne_inf:borne_sup  
    séquence d'instructions  
end
```

Remarque :

Pour utiliser un incrément (pas) autre que 1, on écrit : **borne_inf:pas:borne_sup**



CHAP.V

V.5. Boucle WHILE (tant que ... faire) :

Consiste à effectuer une boucle tant qu'une condition reste vérifiée. On arrête la boucle dès que cette condition n'est plus satisfaite.

Syntaxe :

```
while expression logique  
    séquence d'instructions  
end
```



CHAP.V

APPLICATION 3 (Boucles) :

Soit la somme $S = 16 + \sum_{i=2}^N x^i$.

Calculer S en utilisant deux boucles différentes.

AN : N = 5 et x = 3

