

# Programmation R – Cours 3

Gaëlle LELANDAIS



Version du document : 29/01/2018, ce cours a été conçu  
avec Leslie REGAD

# Boucles et tests

# Boucle « for »

3

- ◉ Principe
  - > Répéter une succession d'instructions
- ◉ Syntaxe générale
  - > for (séquence de répétition) { instructions à répéter }

```
> for (i in 1:6){  
    print(i)  
}  
[1] 1  
[1] 2  
[1] 3  
[1] 4  
[1] 5  
[1] 6
```

```
> comp = 0  
> for (i in seq(5,8)){  
    comp = comp + i  
    print (comp)  
}  
[1] 5  
[1] 11  
[1] 18  
[1] 26
```

# Test « if »

4

## ○ Principe

- > Comparer des éléments
  - Signes comparateurs
    - « > », « < », « == », « <= », « >= », « != »
  - Si le résultat de la comparaison est VRAI les commandes situées entre les accolades sont exécutées

```
> v1 = 15
```

```
> v1 >= 8  
[1] TRUE
```

```
> v1 < 3  
[1] FALSE
```

## ○ Syntaxe générale

- > if (condition) {instruction si « VRAI »} else {instruction si « FAUX »}

```
> a = 3  
> if(a < 3.14){  
    print("c'est vrai »)  
}else{  
    print("c'est faux")  
}  
[1] "c'est vrai »
```

# Conditions multiples

5

## ○ Principe

- > Permettre de combiner plusieurs conditions dans un test
  - *Signes comparateurs*
    - Et « & » ; ou « | | »

```
> Num = 11
> if((Num > 2) & (Num < 10)){
  print("condition ET validee")
}else if((Num > 2) || (Num < 10)){
  print("condition OU validee")
}
```

```
[1] "condition OU validee"
```

# Fonctions

# Ecriture d'une fonction

7

- ⊙ Objectif
  - > Définir ses propres procédures d'analyse
- ⊙ Syntaxe générale
  - > Nom (unique) + paramètres (nommés)

```
nomFonction = function(argument1, argument2){  
    resultat = argument1 + argument2  
    return(resultat)  
}
```

```
> F1 = function(a, b){  
    Op = a + b  
    return(Op)  
}  
> F1(a = 6, b = 20) # appel de la fonction F1  
[1] 26 # valeur renvoyée par la fonction
```

# ⊙ Séance d'exercices