

# TD 4 : ALGORITHMIE

## PROGRAMMATION INFORMATIQUE

---

Morgan Cousin & Joris Falip

16 octobre 2018

Les lignes commentées (#) ne sont pas exécutées par R.

```
# Exemple de multiplication d'une variable par 2
```

```
maVariable <- 8
```

```
maVariable <- maVariable * 2
```

# APPELER FONCTION

- Les fonctions effectuent des actions sur une variable.
- Leur syntaxe est : *action(variable)*

*# Crée un vecteur de valeurs aléatoires entre 0 et 1*

```
vec <- runif(10)
```

*# Donne la longueur d'un vecteur*

```
length(vec)
```

```
## [1] 10
```

*# Calcule la moyenne d'un vecteur*

```
mean(vec)
```

```
## [1] 0.4655358
```

Possibilité de donner des paramètres supplémentaires aux fonctions.

```
# On utilise les arguments pour  
# changer le comportement de runif()  
runif(10, min = 13, max = 26)
```

Une aide est disponible : panneau *help* à droite, ou *?runif*

Possibilité d'utiliser des conditions dans un script.

```
# Si val est inférieur à 0, passe sa valeur à 0
```

```
val <- -5
```

```
if (val < 0) {
```

```
  val <- 0
```

```
}
```

```
val
```

```
## [1] 0
```

Exemples de conditions :

- supérieur >
- inférieur ou égal <=
- strictement égal ==

## ELSE

Condition plus élaborée : si condition alors A, sinon B

```
# Si val pair, j'ajoute 1
# Sinon (val impair) je réduis de 1
val <- 4
if (val %% 2 == 0) {
  val <- val + 1
} else {
  val <- val - 1
}
val

## [1] 5
```

Possibilité d'utiliser plusieurs conditions en même temps avec `&&` et `||`.

```
a <- 0
b <- 10
# si a est nul ET b est supérieur à 99
if(a == 0 && b > 99)
  print("OK")
# si a est nul OU b est supérieur à 99
if(a == 0 || b > 99)
  print("OK")
```

On peut créer une boucle pour répéter les mêmes opérations.

```
vec <- runif(10)
# itérateur prend successivement toutes les valeurs
# de 1 jusqu'à la longueur du vecteur
for(iteateur in 1:length(vec)) {
  print(vec[iteateur])
}
```

# CRÉER FONCTION

Vous pouvez créer vos propres fonctions qui rassemblent plusieurs lignes de code.

```
fonctionPanda <- function(nombre) {  
  nombre <- nombre^2  
  return(sqrt(nombre))  
}
```

```
pourcentageAleatoire <- function(longueur) {  
  return(runif(longueur, min = 0, max = 100))  
}
```