

# MAC0113 - Introdução à Computação para Ciências Humanas

**Aula 15**

---

**Sejam bem-vindas, sejam bem-vindos!**

**R. Hirata Jr.**

# Objetivos de hoje

- Ao final da aula de hoje você deve saber:
  - Vetores
  - Funções

# Vetores (arrays)

- Um vetor é uma sequência de objetos todos do mesmo tipo
- Há três tipos primitivos em R:
  - numérico (numeric) - números
  - alfanumérico (character) - caracteres
  - lógico (logical) - {verdadeiro, falso} - {true, false}
- Pode-se criar vetores de quaisquer tipos primitivos do R
- Uma das formas para se criar um vetor em R é usar o operador de concatenação:
  - `c`
  - Por exemplo
    - `reportCovid = c(4090,2067,530,499,400,364,322,303,298,258,236,200,188)`
    - `newCases =  
c(310822,69300,18873,25642,21469,8790,6796,5547,2897,2880,11472,7723,  
9981)`

# Vetores (arrays)

- Uma vez criado, podemos manipular todos os elementos do vetor usando apenas um nome para quantos valores há no vetor.
- Para isso, precisamos usar o índice do vetor que começa com 1 e termina com o número de elementos do vetor. Além disso, precisamos usar os colchetes ao redor do índice: [ <índice> ] (os sinais de <> são apenas para indicar o que vai dentro dos colchetes).
- Por exemplo:
  - `reportCovid = c(4090,2067,530,499,400,364,322,303,298,258,236,200,188)`
  - `print(reportCovid[1])`
  - 4090
  - `print(reportCovid[13])`
  - 188

# Vetores (arrays)

Para imprimir o vetor inteiro e seus índices, podemos fazer:

```
reportCovid <- c(4090,2067,530,499,400,364,322,303,298,258,236,200,188)
reportCovidn <- length(reportCovid)
print(reportCovidn)
print("O vetor reportCovid contém os elementos:")
i = 1
while (i <= reportCovidn) {
  cat(i,reportCovid[i],"\n")
  i = i + 1
}
```

# Exercício 1

# Dado um vetor  $V$  com uma sequência de números quaisquer, determine se a  
# sequência está em ordem crescente.

```
i = 1
tamanhoV <- length(V)
ehCrescente <- TRUE # indicador de passagem
while ((ehCrescente==TRUE)&(i < tamanhoV)) {
  ehCrescente <- (V[i] < V[i+1])
  i = i + 1
}
if (ehCrescente==TRUE) {
  print("A sequência digitada é crescente.")
} else {
  print("A sequência digitada não é crescente.")
}
```

# Funções em R

- Uma função é um programa como vários que já fizemos, isto é, uma sequência de comandos que implementa algum algoritmo.
- Porém, agora ele estará encapsulado com um nome, um conjunto de parâmetros de entrada e, normalmente, retorna um, ou mais valores.
- Há várias razões para usarmos funções. A principal é organizar nossos programas em pedaços que correspondam a como imaginamos uma solução do problema.

# Funções em R

- Uma função em R é declarada da seguinte maneira:

```
max2 <- function(a,b) {  
  if (a>b) {  
    maxab <- a  
  } else {  
    maxab <- b  
  }  
  return(maxab)  
}
```

**Nome**

**Parâmetros separados por vírgula**

**Retorno da função**

# Funções em R

- Uma vez declarada, basta fazer um carregá-la com source e usar.

```
print(max2(4,5))
```

```
print(max2(5,4))
```

# Exercício 1 - com funções

# Dado um vetor  $V$  com uma sequência de números quaisquer, determine se a sequência está em ordem crescente.

```
ehVetorCrescente <- function(V) {  
  i = 1  
  tamanhoV <- length(V)  
  ehCrescente <- TRUE # indicador de passagem  
  while ((ehCrescente==TRUE)&(i < tamanhoV)) {  
    ehCrescente <- (V[i] < V[i+1])  
    i = i + 1  
  }  
  return(ehCrescente)  
}
```

Obrigado!

---