

Estruturas de Repetição

Prof Diego da Silva Souza

01/06/2021

Estrutura de Repetição

São estruturas que repetem um algoritmo, onde cada ciclo desse algoritmo é denominado *iteração*.

Exemplo: Fazer um bolo simples.

(1) Claras em neves

- **passo 1 (Declaração de objetos):** reunir 3 ovos, 2 tigelas, garfo
- **Passo 2:** pegue o garfo
- **Passo 3:** Pegue 1 ovo
- **Passo 4:** Quebre a casca do ovo com o garfo
- **Passo 5:** Separe a clara da gema e reservá-las em tigelas distintas.
- **Passo 6:** Repetir passos 3, 4 e 5 mais duas vezes
- **Passo 7:** Bater com o garfo as claras na tigela até a consistência estiver semelhante a neve.

Estrutura de Repetição: for

- Sintax:

```
for(object in vector){  
  expression  
}
```

- Exemplo de sintaxes:

Outros Jedis: “Esteja com a força!” Yoda: “Que a força esteja com você!” Jar-Jar-Binks: “Você que esteja a força com!”

Exemplo 01:

```
for(k in c(1,2,3,4,5)){  
  print(k)  
}
```

```
## [1] 1  
## [1] 2  
## [1] 3  
## [1] 4  
## [1] 5
```

Exemplo 02:

```
for(k in c(5,3,1,2,4)){
  print(k)
}
```

```
## [1] 5
## [1] 3
## [1] 1
## [1] 2
## [1] 4
```

Exemplo 03:

```
for(k in 1:10){
  print(k)
}
```

```
## [1] 1
## [1] 2
## [1] 3
## [1] 4
## [1] 5
## [1] 6
## [1] 7
## [1] 8
## [1] 9
## [1] 10
```

Exemplo 04:

```
for(k in c("Leandra","Diego","Thais","Isabella", "Victor", "Celine", "Adriana")){
  print(k)
}
```

```
## [1] "Leandra"
## [1] "Diego"
## [1] "Thais"
## [1] "Isabella"
## [1] "Victor"
## [1] "Celine"
## [1] "Adriana"
```

Exemplo: Fazer um bolo simples.

(1) Claras em neves

- **passo 1 (Declaração de objetos):** reunir 3 ovos, 2 tigelas, garfo Definir **quebrar**(x, y){ z <- x quebrar com y # z = c("casca de x", "clara de x", "gema de x") retorno(z[2:3]) }
- **Passo 2:** for(egg in c(ovo1, ovo2, ovo3)){ *mao_D* <- *garfo* *mao_E* <- **quebrar**(egg,*garfo*) *tigela1* <- *tigela1* + *mao_E*[1] # Acumulador (contadores) *tigela2* <- *tigela2* + *mao_E*[2] # Acumulador (contadores) }

Exemplo 05: Acumuladores

```
acm = 0
for(i in 1:10){
  cat("iteração:", i, " acm inicial:", acm,
      " valor de i:", i,
```

```
    " acm final:", acm + i, "\n")
acm = acm + i
}
```

```
## iteração: 1  acm inicial: 0  valor de i: 1  acm final: 1
## iteração: 2  acm inicial: 1  valor de i: 2  acm final: 3
## iteração: 3  acm inicial: 3  valor de i: 3  acm final: 6
## iteração: 4  acm inicial: 6  valor de i: 4  acm final: 10
## iteração: 5  acm inicial: 10 valor de i: 5  acm final: 15
## iteração: 6  acm inicial: 15 valor de i: 6  acm final: 21
## iteração: 7  acm inicial: 21 valor de i: 7  acm final: 28
## iteração: 8  acm inicial: 28 valor de i: 8  acm final: 36
## iteração: 9  acm inicial: 36 valor de i: 9  acm final: 45
## iteração: 10 acm inicial: 45 valor de i: 10 acm final: 55
```

Exercício para casa: Crie um algoritmo para acumular 100 valores de números aleatórios usando a função *runif(1)*.