



# SLCo610-Introdução a programação

Professor: Seiji Isotani

Professor Assistente: Armando Maciel Toda

Material desenvolvido por: Fernando H. Carvalho

# Conteúdo da Aula

- Revisão da aula passada
- Fluxogramas
- Elementos de laços de repetição *repita n vezes* e *sempre*

# Revisão

- Funções condicionais **SE – ENTÃO**: São estrutura de decisão que permite a escolha do grupo de ações a serem executadas quando determinada **condição** é satisfeita (verdadeira).
- Funções condicionais **SE – ENTÃO - SENÃO**: São estrutura de decisão que permite a escolha entre dois grupos de ações a serem executados dependendo da satisfação de uma condição prévia
- Lembrando que as funções condicionais podem ser **aninhadas** em diversos níveis.

# Exemplo visto em aula: Média e situações

```
quando clicar em   
pergunte Nota da P1: e espere a resposta  
mude n1 para resposta  
pergunte Nota da P2: e espere a resposta  
mude n2 para resposta  
diga junte sua nota é: com  $n1 + n2 / 2$   
se  $n1 + n2 / 2 > 4.9$  então  
  diga Aprovado!  
senão  
  se  $n1 + n2 / 2 > 2.9$  então  
    diga Ficou de REC  
  senão  
    diga Reprovado
```

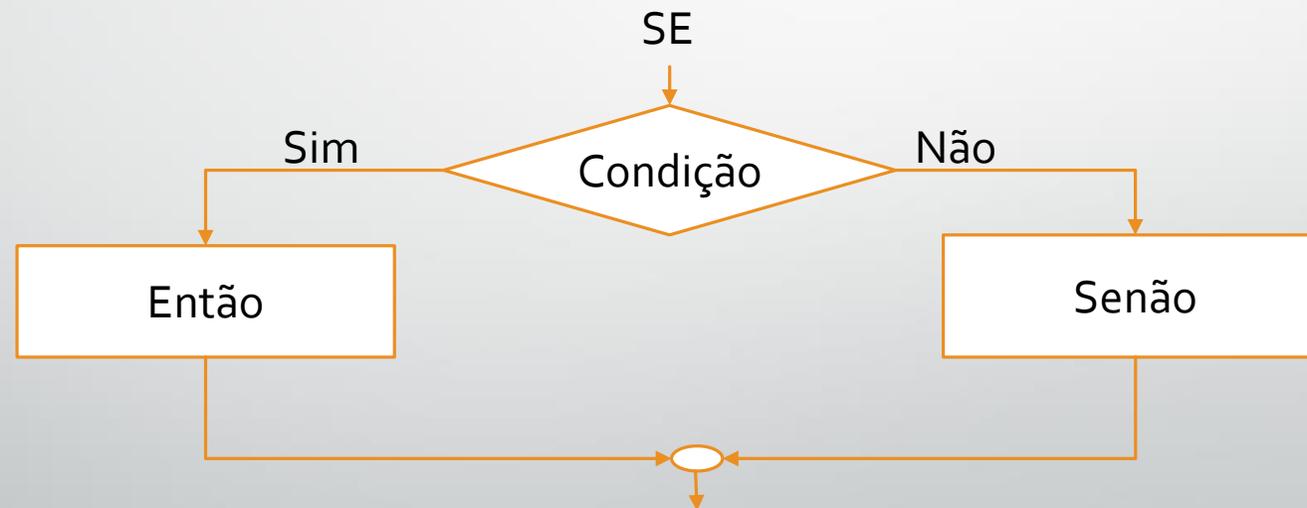
The image shows a Scratch script designed to calculate the average of two grades and determine the student's status based on that average. The script starts with a 'when clicked' event, followed by two 'ask for input' blocks to get 'Nota da P1' and 'Nota da P2'. These inputs are stored in variables 'n1' and 'n2'. A 'say' block then displays the calculated average: 'sua nota é: ' followed by the expression  $n1 + n2 / 2$ . A series of 'if-then-else' blocks follow: first, if the average is greater than 4.9, the student is 'Aprovado!'; otherwise, a second 'if-then-else' block checks if the average is greater than 2.9. If so, the student 'Ficou de REC' (retakes the course); otherwise, the student is 'Reprovado' (failed).

Dúvidas?



# Fluxogramas

- **Fluxogramas** são um tipo de diagrama que pode ser entendido como uma representação esquemática de um processo ou algoritmo

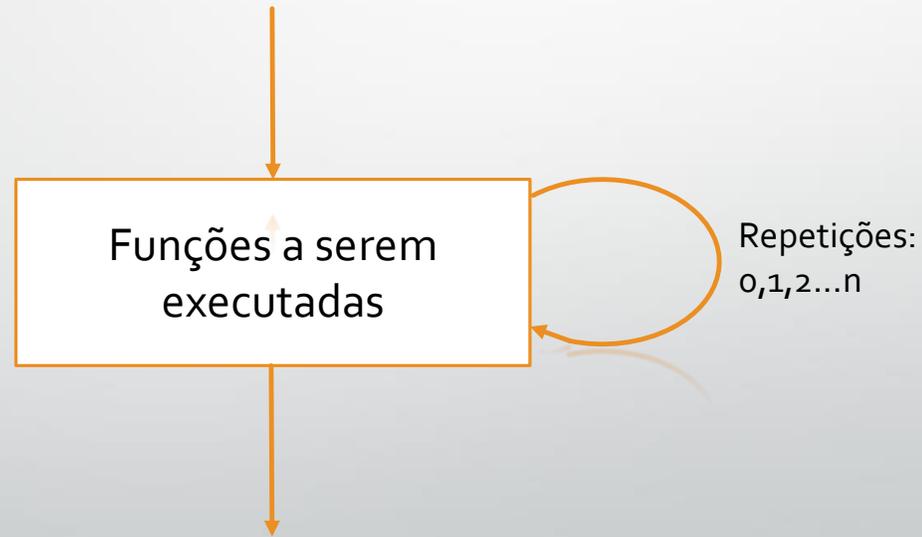


# Exemplos de fluxo



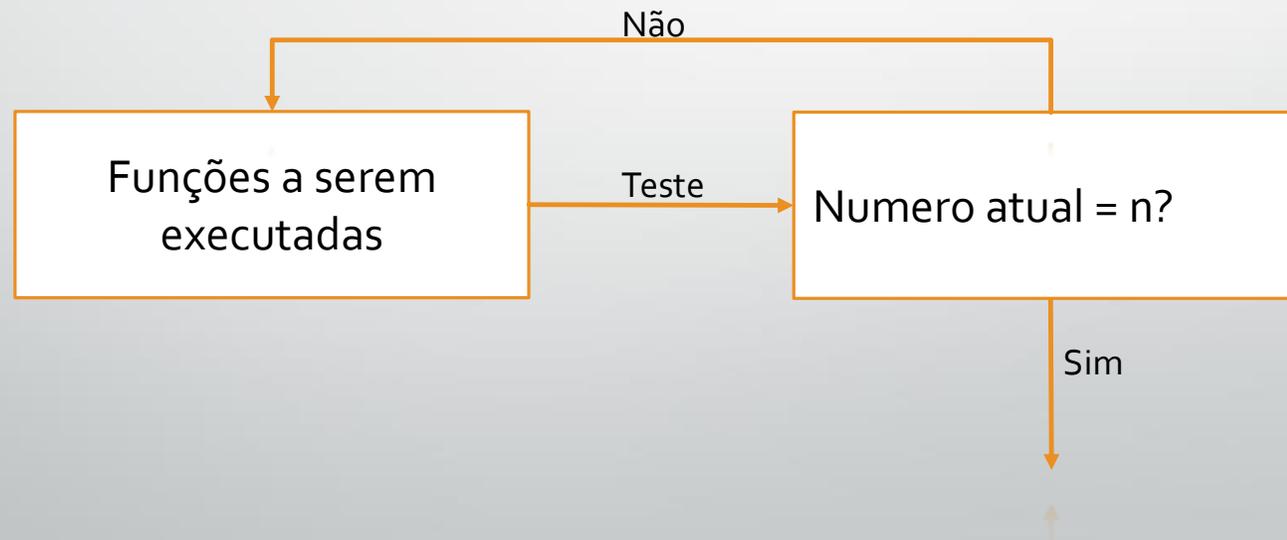
# Elementos de laços de repetição:

- Funções de repetição REPITA N VEZES:
  - São estruturas de laço permitem realizar um grupo de ações n vezes.



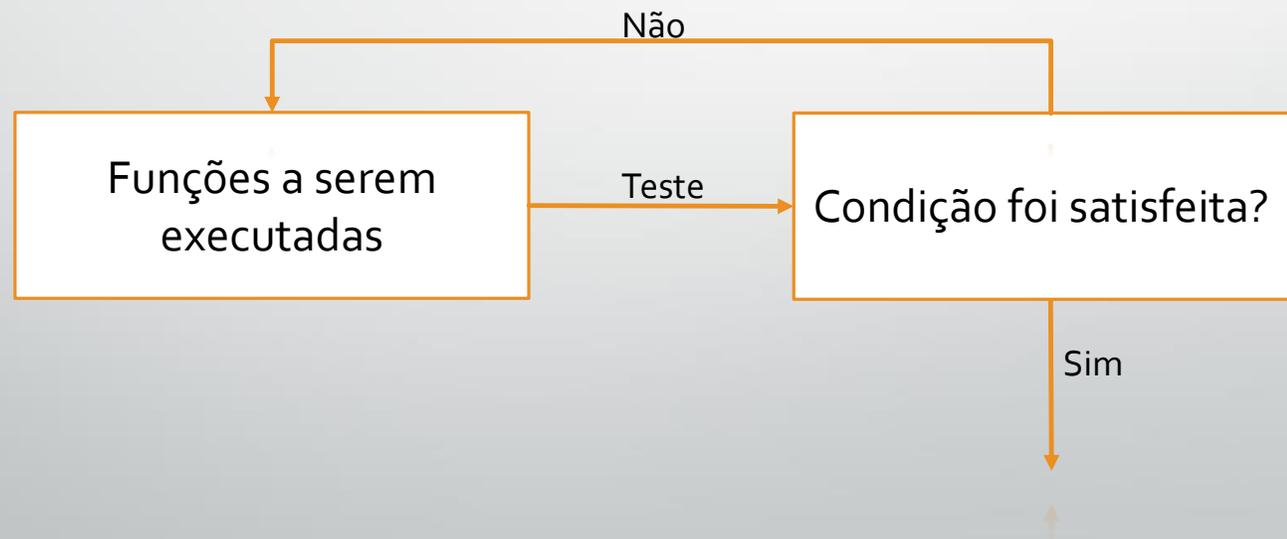
# Elementos de laços de repetição:

- Funções de repetição REPITA N VEZES:
  - São estruturas de laço permitem realizar um grupo de ações n vezes.



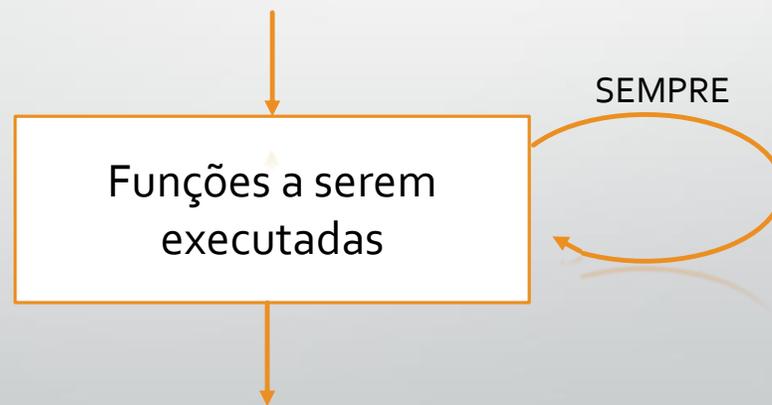
# Elementos de laços de repetição:

- Funções de repetição REPITA ATÉ QUE:
  - São estruturas de laço permitem realizar um grupo de ações um repetidas vezes até que uma condição seja satisfeita.



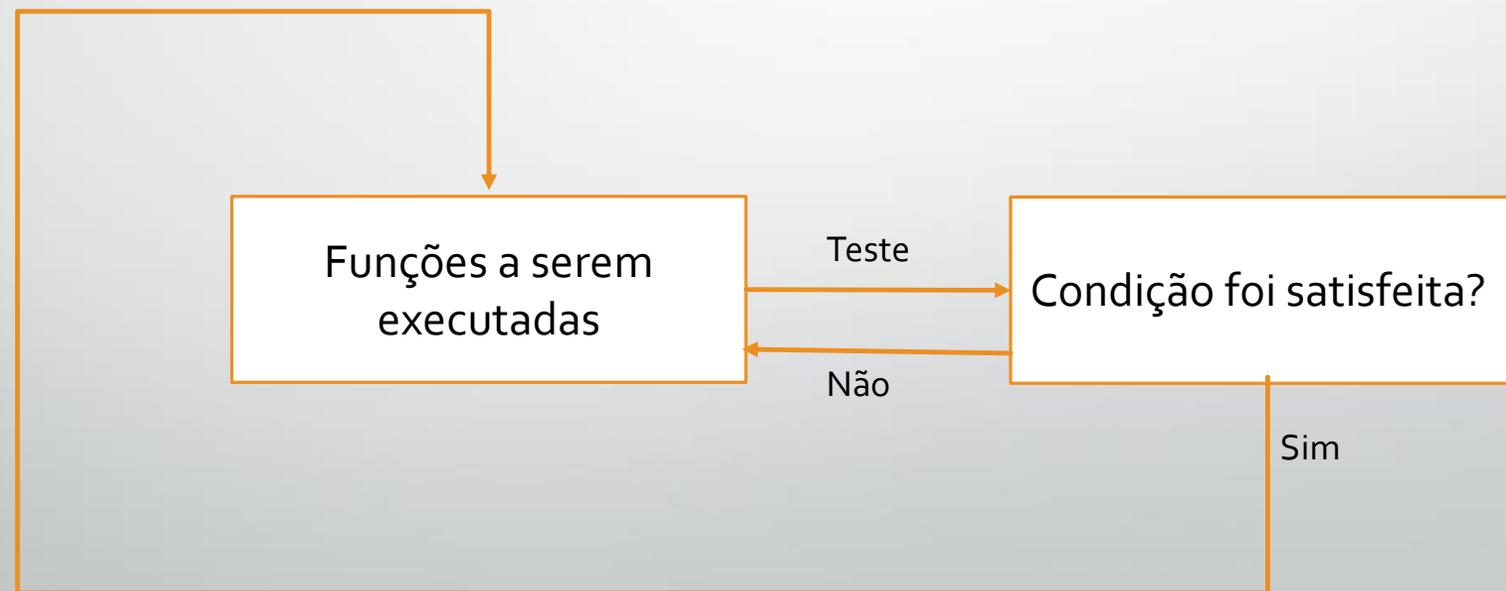
# Elementos de laços de repetição:

- Funções de repetição SEMPRE:
  - São estruturas de laço permitem realizar um grupo de ações repetidas vezes, porém não possuem condições de parada.
  - É necessário utilizar o elemento **parar**, caso contrario seu programa executará infinitamente.



# Elementos de laços de repetição aninhados:

- Assim como os elementos condicionais, elementos de repetição iguais ou diferentes também podem ser aninhados em diversos níveis.



# Exemplos de laços de repetição: Repita n vezes

```
quando clicar em   
  repita 10 vezes  
    mude n para n + 1  
    diga n por 0.5 segundos  
  mude n para 
```

The image shows a Scratch code snippet on a light gray grid background. It starts with an orange 'quando clicar em' block containing a green flag icon. Below it is a yellow 'repita 10 vezes' block. Inside the loop, there are three blocks: an orange 'mude n para n + 1' block, a purple 'diga n por 0.5 segundos' block, and another orange 'mude n para' block with an empty text input field. A white arrow icon is visible at the bottom of the loop block, indicating the loop's continuation.

# Exemplos de laços de repetição: Repita até que



# Exemplos de laços de repetição: Repita n vezes



# Exemplos de laços de repetição aninhados:



Exemplos de laços de repetição em paralelo:



Dúvidas?



# Exercício 1:

- Jogar Detetive!

# Exercício 1:

- Jogar Detetive!
  - Agora façam um fluxograma do jogo Detetive

# Exercício 2:

- Jogo de tabuleiro
  - Como jogar:
    1. Dividam-se em grupos de 4 a 5 pessoas;
    2. Cada jogador começa com 1 carta de cada tipo;
    3. O primeiro jogador rola um dado (D6), e anda um numero de casas igual ao numero tirado no dado, *a próxima casa é sempre seguindo o mesmo caminho que a casa anterior;*
    4. Se a casa for azul ou preta o jogador rola outra vez o dado para saber qual efeito acontece;
    5. Se a casa for laranja, o jogador pode descartar uma carta de função condicional para mudar o caminho atual;
    6. Se a casa for estrela, o jogador escolhe um dos montes de cartas, procura por uma carta que ele deseja, põe na mão e re-embaralha o monte em que procurou;
    7. Cartas descartadas sempre são reembaralhadas em seus respectivos montes;
    8. Ao chegar no final do tabuleiro, se o jogador possui seu objetivo completo ele vence o jogo, caso contrario volta para o inicio do jogo.
  - Observações
    1. Os efeitos das casas azuis, são opcionais;
    2. Os efeitos das casas pretas devem ser realizados ;
    3. O jogador não pode possuir mais do que 1 carta objetivo.



# SLCo610-Introdução a programação

Professor: Seiji Isotani

Professor Assistente: Armando Maciel Toda

Material desenvolvido por: Fernando H. Carvalho