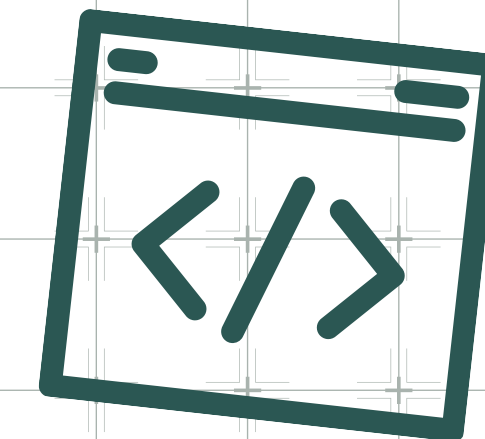
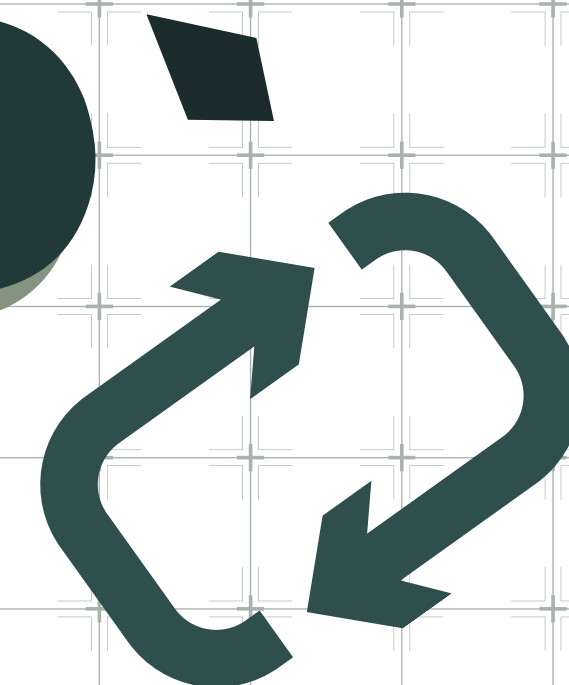


REFORÇO



ESTRUTURAS DE REPETIÇÃO

COMO ESCOLHER E ENTENDER



Lógica de Programação

Objetivos dessa “aula”

- Te ajudar a identificar qual estrutura de repetição é melhor para cada caso;
- Te ajudar a identificar erros comuns em código;
- Te ajudar a entender como elas funcionam.
- Bônus: Comandos não usais

Como elas funcionam?

- Todas as estruturas de repetição possuem um “índice”, que as informa quando elas param.
- Esse índice que a controla e nós ajuda a botar limites, para não ocorrer um loop infinito.



EXEMPLOS

```
Para i de 1 ate 10 faca
```

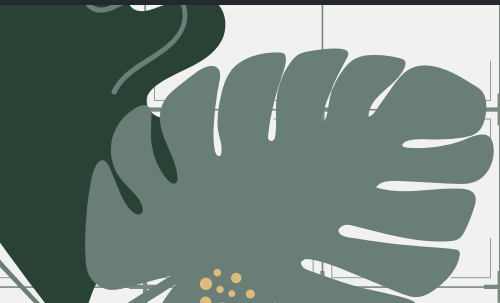
```
enquanto (contador <= 5) faça
```

```
repita
```

```
  escreval("Número: ", contador)
```

```
  contador <- contador + 1
```

```
até (contador > 5)
```



- O i/contador é o nosso “índice” – ele define quando a estrutura para.
- A estrutura enquanto e repita utilizam comparações para definir quando elas precisam parar.
- O índice precisando mudar a cada iteração para não ser infinito, seja por comandos ou pelo input do usuário.

Caso do Para: Vetor

	1	2	3	4
i	39	12	17	26

Quando construindo um vetor, uma linha é criada. O índice, então, é responsável por pegar as “coordenadas” de cada quadro e mostrar o que está preenchido.

Caso do Para: Vetor

$i = 1$

1

2

3

4

i

39

12

17

26

$i = 2$

1

2

3

4

i

39

12

17

26

$i = 3$

1

2

3

4

i

39

12

17

26

$i = 4...$

Caso do Para: Matriz

i	j		
	1	2	3
	4	5	6
	7	8	9
	10	11	12

Uma matriz, por sua vez, é uma tabela com linhas e colunas.

Portanto, o i precisa de um parceiro e normalmente a letra j vem ajudá-lo.

Caso do Para: Matriz

i	j		
	1	2	3
	4	5	6
	7	8	9
	10	11	12

Parecido com coordenadas de Minecraft e pares ordenados de matemática... Cada valor dos dois corresponde à um quadrado. $i = 3, j = 2$;

$i = 2, j = 3$; $i = 4, j = 1$;

SUMÁRIO das diferenças

Para

- Você possui o controle desde o início de quando termina;
- É utilizado também para realizar o input/output de vetores e matrizes;

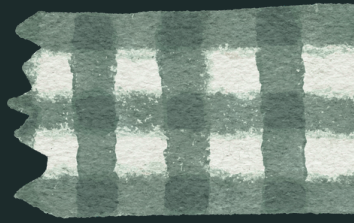
repita

- Começa sem ver se o índice está correto ou não (o usuário pode mudar o número do input dentro do código);

enquanto

- Precisa de passar uma condição de comparação desde o começo. Boa para quando o usuário coloca um número fora do limite, quando você sabe o limite.

EXERCÍCIOS



Slides 11 – 20: Decidir qual estrutura de repetição é a mais adequada para cada cenário.

Gabarito: Slide 21.



Slides 30 – 34: Encontrar o erro escondido em cada erro que impossibilita/atrapalha rodar o código.

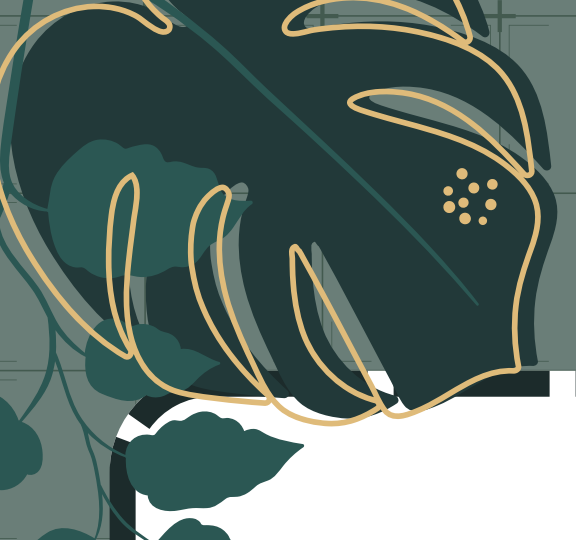
Gabarito: Slide 35 – 37.

Slides 22–25: Interpretar o que cada estrutura de repetição está fazendo, se está certo ou errado.

Gabarito: Slide 26–29.

Qualquer dúvida, reclamação ou sugestão sobre os exercícios, estou de ouvidos abertos!





1: Escolha a melhor estrutura

O cliente gostaria de um programa onde o usuário possa inserir a nota de cada aluno de uma escola. São três notas no total, e há um limite de 10 pontos e um mínimo de 0 pontos. Para fazer esse limite, a estrutura deve ser:

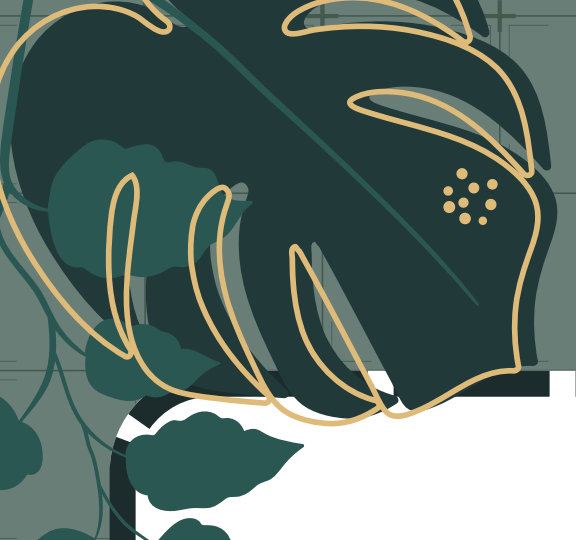
A - Escolha-Caso

B - Enquanto

C - Repita

D - Para





2: Escolha a melhor estrutura

○ cliente gostaria de um programa onde o usuário possa inserir um número de quantidade de pessoas, e preencher o nome de cada uma delas. Para fazer isso, a estrutura deve ser:

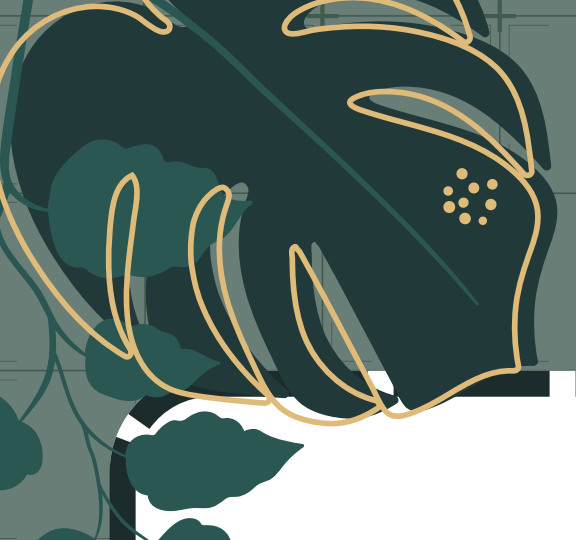
- A - Se/senão
 - B - Enquanto
 - C - Repita
 - D - Para
- 



3: Escolha a melhor estrutura

O cliente gostaria de um programa onde o usuário pode fazer quantas inserções de pessoas ele precisar, uma de cada vez. Para fazer isso, a estrutura deve ser:

- A - Se/senão
 - B - Enquanto
 - C - Repita
 - D - Para
- 



4: Escolha a melhor estrutura

O cliente gostaria de um programa onde, dependendo da escolha do usuário, ele possa mudar de cadastrar um produto, para uma pessoa ou um funcionário. Para fazer isso, a estrutura deve ser:

A - Escolha-Caso

B - Enquanto

C - Repita

D - Para

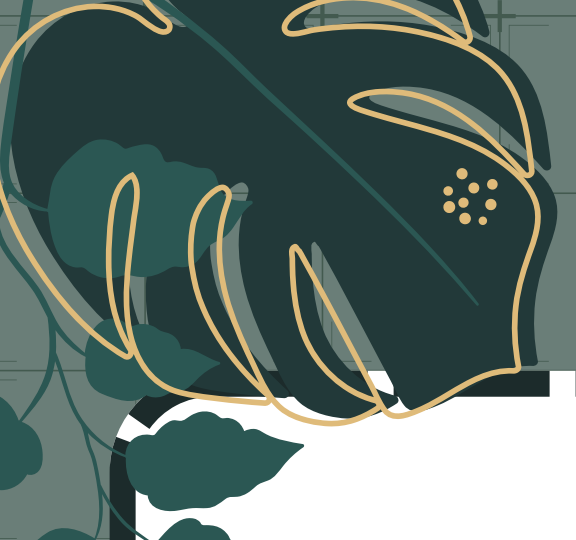




5: Escolha a melhor estrutura

○ cliente gostaria de um programa onde, enquanto cadastrar uma pessoa o obter o seu gênero, contabilizar a quantidade de pessoas que são mulheres ("F") ou homens ("M"). Para fazer isso, a estrutura deve ser:

- A - Se-Senão
 - B - Enquanto
 - C - Repita
 - D - Para
- 



6: Escolha a melhor estrutura

O cliente gostaria de um programa onde sua parte inicial tivesse um “menu” que terminasse o programa apenas quando o usuário inserir certo número.

Para fazer isso, a estrutura deve ser:

- A - Repita
 - B - Enquanto
 - C - Qualquer uma das duas acima
 - D - Para
- 



7: Escolha a melhor estrutura

○ cliente gostaria de um programa onde é inserido o nome de 10 alunos, e então as suas quatro notas. Para fazer isso, a estrutura deve ser:

A - Escolha-Caso

B - Enquanto

C - Repita

D - Para





8: Escolha a melhor estrutura

O cliente gostaria de um programa onde, dependendo da escolha do usuário, ele possa mudar de cadastrar um produto, para uma pessoa ou um funcionário. Para fazer isso, a estrutura deve ser:

- A - Se-Senão
 - B - Enquanto
 - C - Repita
 - D - Para
- 

9: Escolha a melhor estrutura

O cliente gostaria de um programa que calcula, até o usuário inserir o número zero, o MMC do primeiro número que é fixo e não muda com o segundo, que muda a cada iteração. Para fazer isso, a estrutura deve ser:

- A - Enquanto
- B - Enquanto ou Repita
- C - Repita
- D - Para



10: Escolha a melhor estrutura

O cliente gostaria de um programa onde um número é escolhido aleatoriamente, e o usuário tenta os números até conseguir encontrar o número aleatório. Para fazer isso, a estrutura deve ser:

- A - Se-Senão
 - B - Nenhuma das opções
 - C - Repita
 - D - Para
- 

GABARITO

$$a \in B$$

$$A = \{a, b, c\}$$

$$A \cap B$$

1 - B

6 - C

2 - D

7 - D

3 - C

8 - D

4 - A

9 - B

5 - A

10 - C

**Você consegue
justificar por que
uma estrutura foi
escolhida sobre a
outra?**

$$\emptyset$$

$$2^0 = 1$$

1 - Interpretação

Início

```
// Seção de Comandos, procedimento, funções,  
escreval ("Digite o número do mês:")  
leia (m)  
se m = (1) entao  
    escreval ("Esse mês possui 31 dias.")  
fimse  
se m = (2) entao  
    escreval ("Esse mês possui 28 dias.")  
fimse  
se m = (3) entao  
    escreval ("Esse mês possui 31 dias.")  
fimse  
se m = (4) entao  
    escreval ("Esse mês possui 30 dias.")  
fimse  
se m = (5) entao  
    escreval ("Esse mês possui 31 dias.")  
fimse  
se m = (6) entao  
    escreval ("Esse mês possui 30 dias.")  
fimse  
se m = (7) entao  
    escreval ("Esse mês possui 31 dias.")  
fimse  
se m = (8) entao  
    escreval ("Esse mês possui 31 dias.")  
fimse
```

```
se m = (9) entao  
    escreval ("Esse mês possui 30 dias.")  
fimse  
se m = (10) entao  
    escreval ("Esse mês possui 31 dias.")  
fimse  
se m = (11) entao  
    escreval ("Esse mês possui 30 dias.")  
fimse  
se m = (12) entao  
    escreval ("Esse mês possui 31 dias.")  
fimse
```

Seria melhor utilizar uma
outra estrutura de
repetição, como
escolha/caso?

2 - Interpretação

Início

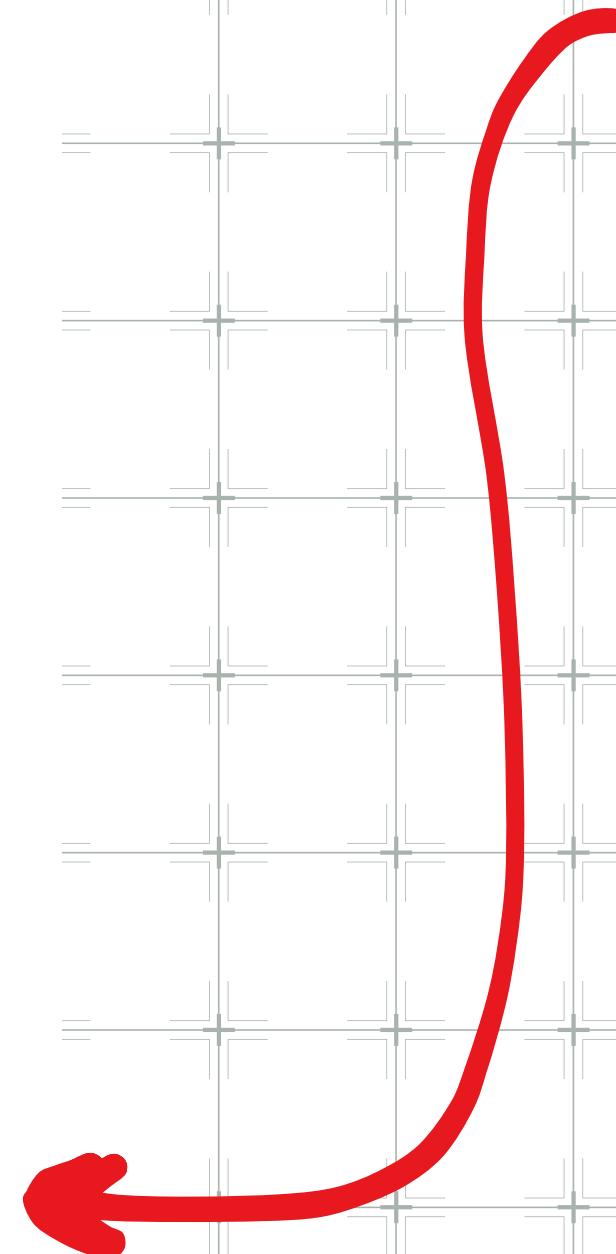
```
// Seção de Comandos, procedimento, funções,  
para i de 1 até 5 faça  
  escreva("digite o",i,"* número de A: ")  
  leia(a[i])  
  b[i] <- a[i]*a[i]  
fimpara  
escreval("=====  
para i de 1 até 5 faça  
  escreval("Vetor A,",i,"* número:",a[i])  
fimpara  
escreval("=====  
para i de 1 até 5 faça  
  escreval("Vetor B,",i,"* número:",b[i])  
fimpara
```

O que está sendo
construído nesse código?
(Vetor, para)

3 - Interpretação

```
para i de 1 ate 5 faca
  escreva("Insira os números: ")
  leia(v[i])
fimpara
limpatela
escreval("Números           : |")
para i de 1 ate 5 faca
  escreva(v[i],",", " ")
fimpara
escreval("           ")
para i de 1 ate 5 faca
  para j de 1 ate 5 faca
    se v[i] < v[j] entao
      aux <- v[i]
      v[i] <- v[j]
      v[j] <- aux
    fimse
  fimpara
fimpara
para i de 1 ate 5 faca
  escreva(v[i],",", " ")
fimpara
```

Como estará o vetor no
último para?
Dica: Faça o teste de mesa
com menos espaços no
vetor!



4 - Interpretação

```
se (i mod 2 = 0) entao  
  [i] <- numeros[i]  
fimse
```

Quais números essa
estrutura está
selecionando?

1 - Interpretação

```
Var
// Seção de Declarações
m: inteiro

Inicio
// Seção de Comandos, procedimento, funções,
// chamadas de procedimentos e funções
escreval ("Digite o número do mês:")
leia m
// 40
escreval ("Esse mês possui 31 dias.")
// 41
fimescolha
// 49
escreval ("Esse mês possui 31 dias.")
// 50
fimse

caso (4)
    escreval ("Esse mês possui 30 dias.")
caso (5)
    escreval ("Esse mês possui 31 dias.")
caso (6)
    escreval ("Esse mês possui 30 dias.")
caso (7)
    escreval ("Esse mês possui 31 dias.")
caso (8)
    escreval ("Esse mês possui 31 dias.")
caso (9)
    escreval ("Esse mês possui 30 dias.")
caso (10)
    escreval ("Esse mês possui 31 dias.")
caso (11)
    escreval ("Esse mês possui 30 dias.")
caso (12)
    escreval ("Esse mês possui 31 dias.")
fimescolha
```

Há uma linha de comentário à mais, por isso 41!

Sim! Embora não tenha nada de errado, você salva 10 linhas (esforço, e tempo) com escolha/caso!

2 - Interpretação

Início

```
// Seção de Comandos, procedimento, funções,  
para i de 1 até 5 faça  
  escreva("digite o",i,"* número de A: ")  
  leia(a[i])  
  b[i] <- a[i]*a[i]  
fimpara  
escreval("=====  
para i de 1 até 5 faça  
  escreval("Vetor A,",i,"* número:",a[i])  
fimpara  
escreval("=====  
para i de 1 até 5 faça  
  escreval("Vetor B,",i,"* número:",b[i])  
fimpara
```

Está sendo construído
outro vetor com o
quadrado do número do
vetor A!

3 - Interpretação

```
para i de 1 ate 5 faca
  escreva("Insira os números: ")
  leia(v[i])
fimpara
limpatela
escreval("Números na ordem de entrada: |")
para i de 1 ate 5 faca
  escreva(v[i],",", " ")
fimpara
escreval(" ")
escreval("Ordenado de forma crescente: ")
para i de 1 ate 5 faca
  para j de 1 ate 5 faca
    se v[i] < v[j] entao
      aux <- v[i]
      v[i] <- v[j]
      v[j] <- aux
    fimse
  fimpara
fimpara
para i de 1 ate 5 faca
  escreva(v[i],",", " ")
fimpara
```

Estará organizado de
forma crescente!

4 - Interpretação

Apenas os números
pares!

```
se (i mod 2 = 0) entao  
  [i] <- numeros[i]  
fimse
```

1: Encontre o erro!

```
enquanto n >= 0 e n <= 10 faça
  escreval ("Qual é a primeira nota do aluno?")
  leia(n)
  escreval("Digite a segunda nota:")
  leia(n1)
  n2<- n1 + n
  escreval("A média é:",n2/2)
fimenquanto
```

$A \cap B$

$a \in A$

$\emptyset$

$A = f$



2: Encontre o erro!

```
Var
    // Seção de Declarações das variáveis
    numeros: vetor[1..10] de inteiro
    i: inteiro
Inicio
    // Seção de Comandos, procedimento, função
    para i de 1 ate 10 faca
        escreva("Digite o ", i, "º número: ")
        leia(numeros[i])
    fimpara
    limpatela
    escreval("====Conteúdo do Vetor:====")
    para i de 1 ate 10 faca
        escreva(i, "º número: ")
        escreval(numero[i])
    fimpara
```

3: Encontre o erro!

Var

// Seção de Declarações das variáveis

sal: inteiro

j: real

Início

escreval("Insira seu salário: ")

leia(sal)

j <- sal * 1.5%

$A \cap B$

$a \in A$

$\emptyset$

$A = f$



4: Encontre o erro!

```
se n[i] < 0 entao  
    r = r + 1  
fimse
```

$A \cap B$

$a \in A$

$\emptyset$

$A = f$



1: Encontre o erro!

```
enquanto n >= 0 e n <= 10 faça  
    escreva "Digite a primeira nota do aluno?"  
    leia n1  
    escreva "Digite a segunda nota:"  
    leia n2  
    escreva "A média é:", n2/2)  
fimenquanto
```

Aviso

VisuAlg3 encontrou o seguinte problema em seu algoritmo, na linha 15:

Esperava encontrar FACA.

Conteúdo da linha:

```
enquanto n >= 0 e n <= 10 faça
```

$A \cap B$
 $a \in B$

$\emptyset$

$A = f$



2: Encontre o erro!

```
Var
    // Seção de Declarações das variáveis
    numeros: vetor[1..10] de inteiro
    i: inteiro
Inicio
    // Seção de Comandos,
    para i de 1 ate 10 fa
        escreva("Digite o
        leia(numeros[i])
    fimpara
    limpatela
    escreval("====Conteúdo====")
    para i de 1 ate 10 faca
        escreva(i,"° número: ")
        escreval(numero[i])
    fimpara
```

VisuAlg3 encontrou o seguinte problema em seu algoritmo, na linha 21 :

Variável 'NUMERO[1]' não foi encontrada.

Conteúdo da linha:

```
escreval(numero[i])
```

3: Encontre o erro!

```
Var  
    // Seção de  
    sal: inteiro  
    j: real  
Inicio  
escreval("Insira o valor de sal")  
leia(sal)  
j <- sal * 1.5%
```

VisuAlg3 encontrou o seguinte problema em seu algoritmo, na linha 16 :

Sintaxe incorreta.

Conteúdo da linha:

```
j <- sal * 1.5%
```

```
j <- sal * 1.5/100
```

$A \cap B$
 $a \in A$

$\emptyset$

$A = f$



4: Encontre o erro!

```
se n[i] < 0  
    r = r + 1  
    r <- r + 1
```

VisuAlg3 encontrou o seguinte problema em seu algoritmo, na linha 4:
Sintaxe incorreta.

$A \cap B$

$a \in A$

$\emptyset$

$A = f$



limpatela – Limpa a tela antes do comando

mod – divide o número até um valor

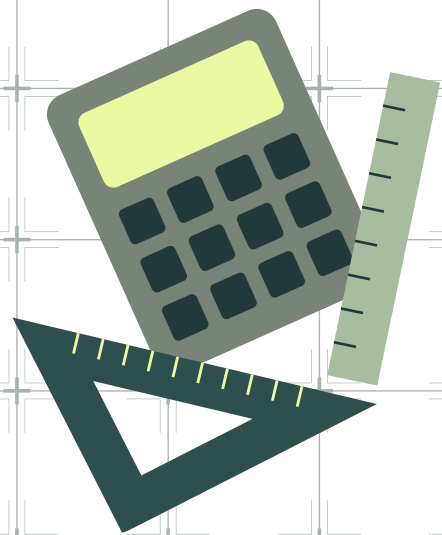
aleatorio – coloca um número aleatório na variável, com limites (equivalente à random de python)

interrompa – interrompe uma estrutura de repetição (equivalente à break de muitas linguagens)



OBRIGADA

Que esses exercícios tenham te ajudado!



$$a \div b \text{ or } \frac{a}{b}$$