**ANOTACOES – 02/04/24**

**CONTINUACAO OPERADORES**

**OPERADORES DE IDENTIDADE**

Os operadores de identidade servem para a comparação de objetos

**Operadores:**

- Is

- Is not

**Exemplo:**

**IS:** Nome **is** “Kaue”

**IS NOT:** x **is not** “Python”

**ATIVIDADE:**

X1= “Botafogo”

X2 =” Flamengo”

X3 =” Botafogo”

**1) Xresult = x1 is x3 = Verdade**

**2) Xresult = x1 is not x2 = Verdade**

**3) Xresult = X1 is x3 = Verdade**

**4) Xresult = x2 is not x3= Verdade**

**OPERADORES DE ASSOCIACAO**

Você basicamente vai calcular se algo pertence ou não a uma variação, eles servem para comparação de objetos, nessa comparação e verificado se eles ocupam a mesma posição na memória.

**Operadores:**

**IN = 2 in x**

**NOT IN= 2 not in x**

**Exemplo:**

*frutas = ["banana","laranja","uva","ameixa"]*

*fruta\_1 = "ameixa"*

*fruta\_2 = "melancia"*

*ESCREVA (fruta\_1 in frutas) # True*

*ESCREVA (fruta\_2 in frutas) # False*

**ATIVIDADE:**

**Xresult = x in frutas = Verdadeiro**

**Xresult = x not in frutas = Falso**

**Xresult = x2 not in frutas = Falso**

**Xresult = x3 in frutas = Falso**

**ATIVIDADES:**

**X = 4**

**X1 = 9**

**X2 = 5**

**XResult = (((2 + 5) \* 3) + X + 5) < X1**

**XResult = (((2 + 5) \* 3) + (X + 5 < X1) == (X == 6 + X2 \* 2))**

**XResult = (((3 \* 6) + X - 3) < X2)**

**XResult = (((7 \* 8) > (X + 4 > X1) < (X! = 2 + X2 \* 9)))**

**LOGICA DE PROGRAMACAO**

**OPERADORES CONDINCIONAIS**

**TIPOS DE ESTRUTURAS**

Dependendo do conjunto de dados de entrada do algoritmo, deve-se executar um conjunto diferente de instruções, sendo necessário controlar o fluxo de execução das instruções (a sequência em que as instruções são executadas) em função dos dados fornecidos como entrada ao mesmo e outras condições.

**ESTRUTURA SEQUENCIAL**

comandos enviados numa sequência pré-estabelecida. Cada comando e executado somente após o termino do comando anterior

**exemplo:**

leia (a)

**y = a \* a**

**escreva (y)**

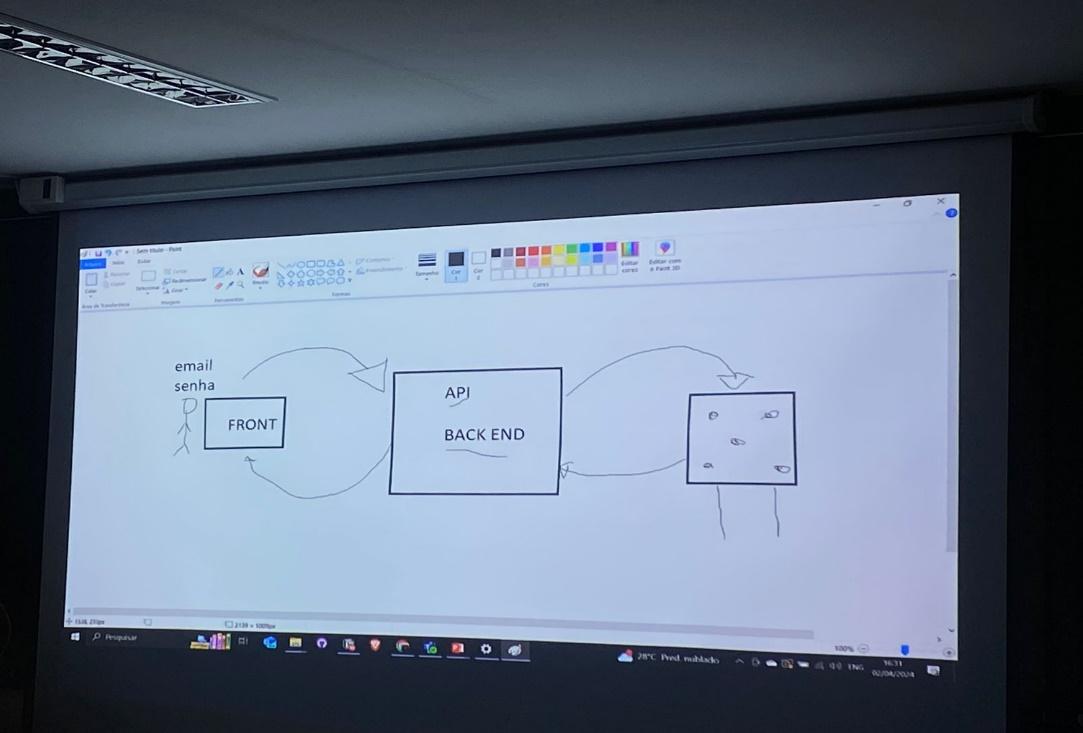
**ESTRUTURA DE DESICAO**

Usaremos com constância, para a decisões até mesmo no cotidiano, decisões básicas que vão ate de abrir o olho de manhã até decisões importantes e necessárias.

Isso está instalado no nosso sistema, tomamos decisões naturalmente oque podemos chamar de intuição.

**Estrutura de decisão na programação**

Ao entrar em uma rede social, você precisa informar seus dados de e-mail e senha e o código verifica se essas informações são validas

****

**BELO DESENHO DO PROFESSOR CAIQUE:** para explicar a trajetória de uma estrutura de decisão na programação.

**ESTRUTURA DE DESICAO NA PROGRAMACAO**

O "if" e "else" são funções condicionais que dependem de uma condição que impusermos a elas. Podemos definir essa estrutura, ao mesmo tempo que representamos em um exemplo, da seguinte

forma:

Se ("if") um número é maior que 30, então retornará a palavra

"maior", se não ("else"), retornará “menor".

**Exemplo**

**If (expressão\_for\_verdadeira):**

Executar\_primeiro\_bloco\_de\_codigo ()

Elif comando utilizado quando queremos realizar a verificação de outra expressão caso a primeira validação seja falsa

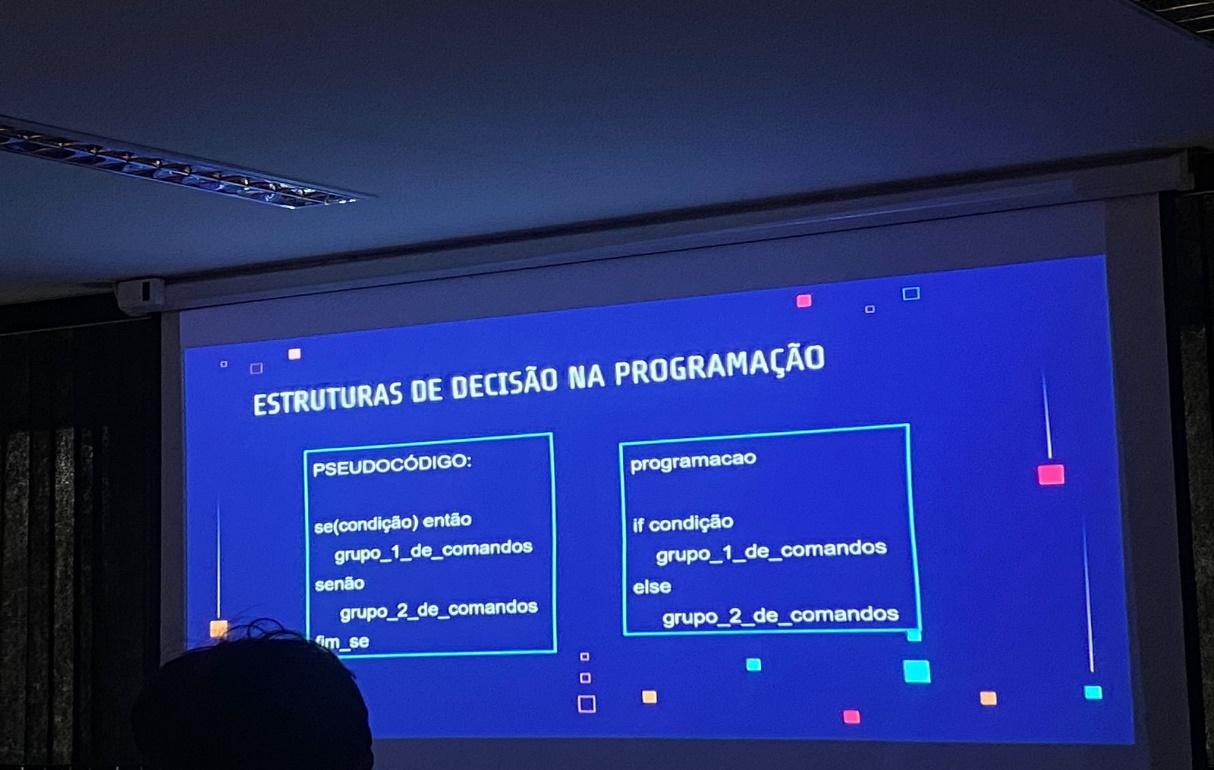
Traduzido para “senão se”.

If (expressao\_for\_verdade)

(executar\_algo)

Elif (expressao2\_for\_verdade)

(executar\_outro\_algo)



**ATIVIDADE:**

