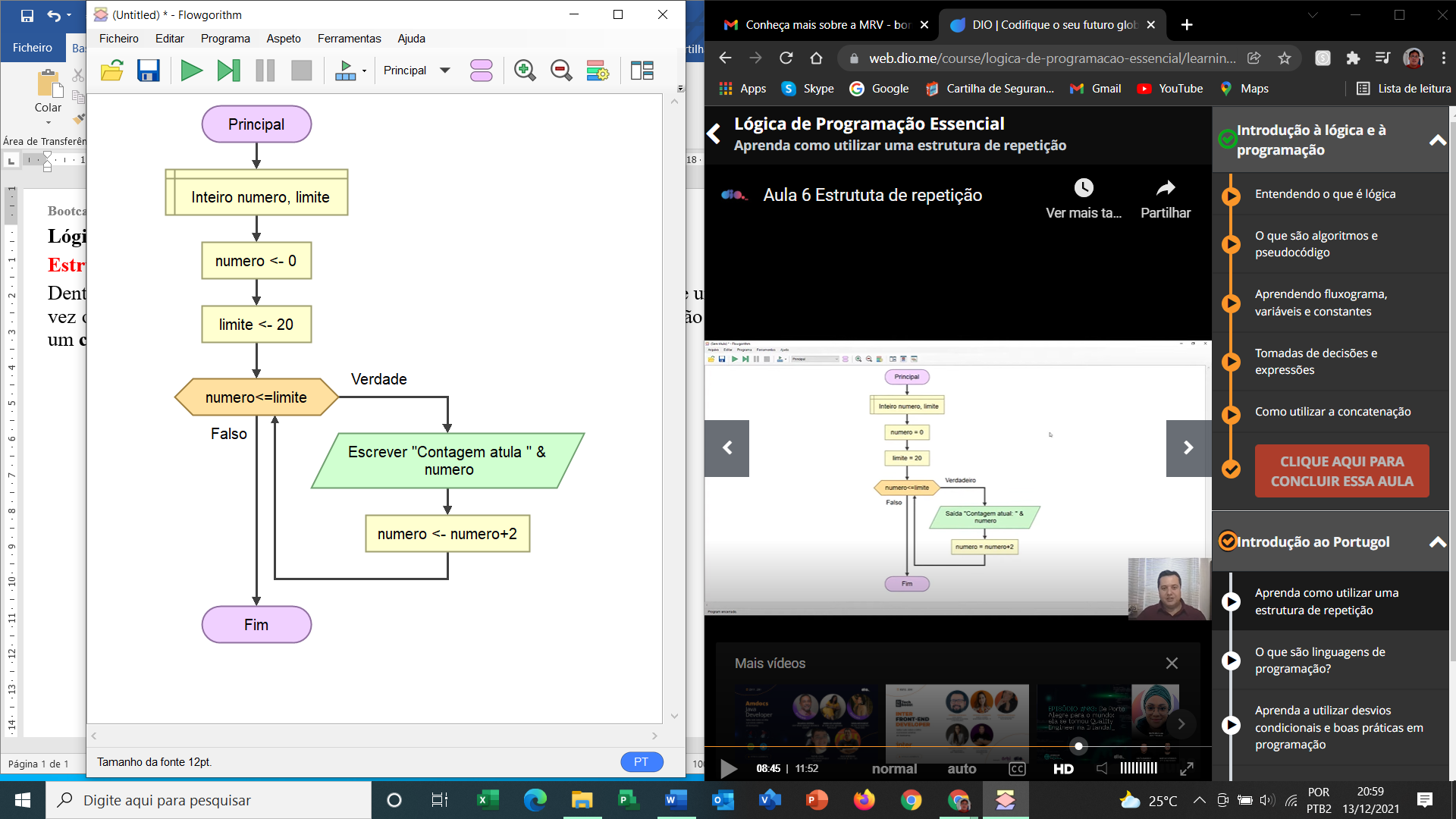
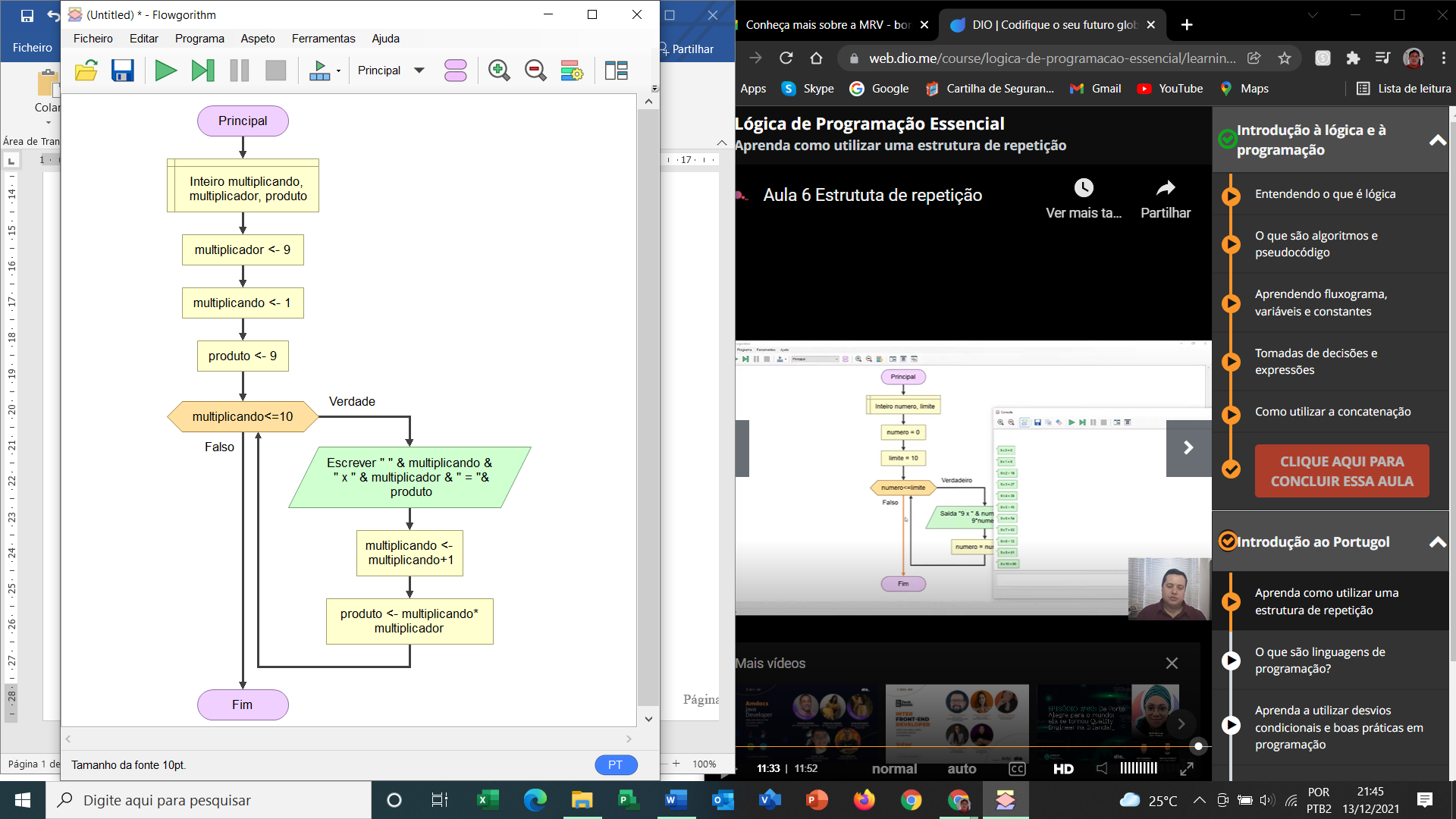
**Lógica de Programação Essencial - Introdução ao Portugol**

**Estrutura de repetição**

Dentro da lógica de programação é uma estrutura que permite executar mais de uma vez o mesmo comando ou conjunto de comandos, de acordo com uma condição ou um **contador**.



**Desafio:** Fluxograma da tabua do 9, usando estrutura de repetição



**Linguagem de programação e Portugol**

**Linguagem de Alto nível:** são aquelas cuja sintaxe se aproxima mais da nossa linguagem e se distanciam mais da linguagem de máquina

**Linguagem de Baixo nível:** são aquelas que se aproximam mais da linguagem de máquina. Essas são as linguagens que você precisa ter o conhecimento direto da arquitetura do computador para fazer alguma coisa.

**Ligugagens Compiladas:** é uma linguagem de programaçao em que o código fonte, é executado diretamente pelo sistema operacional ou pelo processador, após ser traduzido por meio de um processo chamado compilação.

**Linguagens Interpretadas:** são linguagens de programação em que o código fonte é executado por um programa de computador chamado interpretador, que em seguida é executado pelo sistema operacional ou processador.

**Portugol**

Portugol é uma pseudolinguagem que permite ao leitor desenvolver algorítmos estruturados em português de forma simples e intuitiva, independentemente de uma linguagem de programação.

Éuma pseudolinguagem que permite ao programador pensar no problema em si e não no equipamento que irá executar o algorítmo.

Link para baixar o portugol:

<https://github.com/UNIVALI-LITE/Portugol-Studio/releases/>

**Exemplo 1: Média do aluno**

programa

{

funcao inicio()

{

real nota1,nota2,nota3,nota4,media

cadeia aluno //cadeia = conjunto de caracteres

escreva("Digite o nome do aluno: ")

leia(aluno)

escreva("Digite a nota 1: ")

leia(nota1)

escreva("Digite a nota 2: ")

leia(nota2)

escreva("Digite a nota 3: ")

leia(nota3)

escreva("Digite a nota 4: ")

leia(nota4)

media = (nota1+nota2+nota3+nota4)/4

escreva("O aluno: " + aluno + " obteve a média: " + media)

//+ = faz a concatenação no portigol

}

}

**Desafio: Venda total e a média durante quatro meses**

programa

{

funcao inicio()

{

real vendaJan,vendaFev,vendaMar,vendaAbr,vendaTotal,media

cadeia vendedor

escreva("Informe o nome do vendedor: ")

leia(vendedor)

escreva("Informe o valor da venda no mês de janeiro: ")

leia(vendaJan)

escreva("Informe o valor da venda no mês de fevereiro: ")

leia(vendaFev)

escreva("Informe o valor da venda no mês de março: ")

leia(vendaMar)

escreva("Informe o valor da venda no mês de abril: ")

leia(vendaAbr)

vendaTotal = vendaJan+vendaFev+vendaMar+vendaAbr

media = vendaTotal/4

escreva("A venda total foi de: " + vendaTotal + " e a média foi: " + media)

}

}

**Desvios Condicionais e boas práticas de programação**

**Desvio Condicional - se-senao:**

“É utilizada a palavra reservada **se**, a condição a ser testada entre parênteses e as instruções que devem ser executadas entre chaves caso o desvio seja **verdadeiro**”

Caso a condição seja falsa um outro conjunto de comandos deve ser executado, o **senao**.

//Programa calcula a media aritmética

//Autor: Bonifacio

programa

{

funcao inicio()

{

real nota1,nota2,nota3,nota4,media

cadeia aluno //cadeia = conjunto de caracteres

escreva("Digite o nome do aluno: ")

leia(aluno)

escreva("Digite a nota 1: ")

leia(nota1)

escreva("Digite a nota 2: ")

leia(nota2)

escreva("Digite a nota 3: ")

leia(nota3)

escreva("Digite a nota 4: ")

leia(nota4)

media = (nota1+nota2+nota3+nota4)/4

escreva("O aluno: " + aluno + " obteve a média: " + media)

//+ = faz a concatenação no portigol

se(media>=7) {

escreva("\n" + "Parabéns! Você foi aprovado")

}

senao {

escreva("\n" + "Infelizmente você foi reprovado")

}

}

}

// Programa usando se

// Autor: Bonifacio

programa

{

funcao inicio()

{

escreva("1 - Abrir Netflix 2 - Abrir Amazon Prime 3 - Abrir HBO GO 4 - sair")

inteiro menu = 0

escreva("\n" + "Sua opção")

leia(menu)

se (menu==1) {

escreva("OK!! Abrir Netflix!!")

}

se (menu==2) {

escreva("OK!! Abri Amazon Prime!!")

}

se (menu==3) {

escreva("OK!! Abrir HBO GO!!")

}

se (menu==4) {

escreva("Saindo do menu...")

}

}

}

**Desvio condicional - caso**

Este comando é similar aos comandos **se** e **senão**, e reduz a complexidade na escolha de diversas opções. Apesar de suas similaridades com o **se**, ele possui algumas diferenças. Neste comando não é possível o uso de operadores lógicos, ele apenas trabalha com valores definidos.

// Programa usa do escolha + caso

// Autor: Bonifacio

programa

{

funcao inicio()

{

escreva("Escolha uma das opções: 1 - Abrir Netflix 2 - Abrir Amazon Prime 3 - Abrir HBO GO 4 - Sair")

inteiro menu=0

escreva ("\n" + "Sua escolha: ")

leia (menu)

escolha (menu)

{

caso 1: // testa se o valor é igual a 1

escreva ("OK! Abrir Netflix!")

pare

caso 2: // testa se o valor é igual a 2

escreva ("OK! Abrir Amazon Prime!")

pare

caso 3: // testa se o valor é igual a 3

escreva ("OK! Abrir HBO GO!")

pare

caso 4:

escreva ("Saindo do menu...")

pare

caso contrario:

escreva ("Você deve escolher as opções 1, 2, 3 ou 4")

}

}

}

**Laços de repetição - Portugol**

// Programa para calcular uma tabuada

// Autor: Bonifacio

programa

{

funcao inicio()

{

inteiro contador,limite,resultado,numero

escreva ("Digite um número: ")

leia(numero)

escreva ("Digite o limite máximo da tabuada: ")

leia(limite)

contador = 0

faca{

resultado = numero \* contador

escreva(numero + " x " + contador + " = " + resultado + "\n")

contador ++ // contador = contador + 1

}enquanto (contador<=limite)

}

}

**Matrizes e vetores**

Uma **matriz** é uma coleção de variáveis de mesmo tipo, acessíveis com um único nome e armazenados contiguamente na memória.

A individualização de cada variável de um vetor é feita através de uso de **índices**.

Os **vetores** são matrizes de uma só dimensão

**Exemplo**

cadeia Vetor[5]; // declara um vetor de 5 posições

cadeia Matriz[5][3]; // declara uma matriz de 5 linhas e 3 colunas

cadeia fruta[4];

frutas[0]=”Maçã”

frutas[1]=”Pera”

frutas[2]=”Uva”

frutas[3]=”Melão”

escreva(frutas[2])

cadeia cesta[][] = {{“Maça”, “100”},{“Pera”,”200”},{“Melão”,”300”}}

// Exemplo de Vetor

// Autor: Bonifacio

programa

{

funcao inicio()

{

cadeia frutas[4]

inteiro contador = 0

frutas[0]="Maçã"

frutas[1]="Pera"

frutas[2]="Uva"

frutas[3]="Jaca"

faca{

escreva(frutas[contador] + "\n")

contador++

}enquanto (contador<=3)

}

}

// Exemplo de Matriz

// Autor: Bonifácio

programa

{

funcao inicio()

{

inteiro contador=0

cadeia cesta[][]={{"Pera","100"},{"Jaca","200"},{"Maça","900"},{"Uva","89"}}

// escreva ("Produto: ")

// escreva (cesta[0][0])

// escreva (" Quantidade: ")

// escreva (cesta[0][1])

// exibir a tabela inteira

faca{

escreva("Produto: " + cesta[contador][0] + " Quantidade: " + cesta[contador][1] + "\n")

contador++

}enquanto(contador<=3)

}

}

**Exercício final**

Crie uma matriz para armazenar e exibir as seguintes informações:

|  |  |  |
| --- | --- | --- |
| Nome | Cidade | Telefone |
| João | São Paulo | (11) 9999-5241 |
| Maria | Ribeirão Preto | (16) 9999-8596 |
| Ana | Manaus | (92) 9999-8574 |

// Fazer um programa para exibir as informações de uma matriz

// Autor: Bonifácio

programa

{

funcao inicio()

{

inteiro contador=0

cadeia nome[] = {"João","Maria","Ana"}

cadeia cidade\_fone[][]={{"São Paulo","(11) 9999-5241"},{"Ribeirão Preto","(16) 9999-8596"},{"Manaus","(92) 9999-8574"}}

faca{

escreva("Nome:" + nome[contador] + " >>" + " Cidade:" + cidade\_fone[contador][0] + " >>" + " Telefone:" + cidade\_fone[contador][1] + "\n")

contador++

}enquanto(contador<=2)

}

}