# AT\_16\_Exercícios MistosAT\_16\_Exercícios Mistos

# 

1\_Contador de Vogais

Crie um programa que leia uma palavra e utilize a biblioteca Texto para contar quantas vogais existem nela. Ao final, exiba o total de vogais.

programa {

inclua biblioteca Texto --> t

funcao inicio() {

cadeia palavra

inteiro qtd, i, vogais =0

caracter indice, teste

escreva("Escreva uma palavra: ")

leia(palavra)

qtd = t.numero\_caracteres(palavra)

para (i = qtd -1; i>=0; i--){

indice = t.obter\_caracter(palavra, i)

se (indice == "a" ou indice == "e" ou indice == "i" ou indice == "o" ou indice == "u"){

vogais++

}

}

escreva("O total das vogais é: ", vogais)

}

}

2\_ Tabuada com Escolha

Desenvolva um programa que peça ao usuário um número entre 1 e 10 e utilize a estrutura PARA...ATE...FACA para exibir sua tabuada. Valide a entrada com estrutura de decisão.

programa {

funcao inicio() {

inteiro num = 0, tab,calc

faca{

escreva("Digite o número que será feita a tabuada: ")

leia(num)

}enquanto(num<=0 ou num>10)

para(inteiro tab = 1; tab<=10; tab++){

calc = tab \*num

escreva(tab, "x",num,"=", calc, "\n")

}

}

}

3\_Sorteador até Acertar

Usando a biblioteca UTIL, sorteie números de 1 a 10 até que o número 7 seja sorteado. Exiba todos os números sorteados. Use a estrutura ENQUANTO.

programa {

inclua biblioteca Util --> u

funcao inicio() {

inteiro certo = 7, num, tentativas = 0

enquanto ( num!= certo){

num = u.sorteia(1, 10)

tentativas++

escreva ("\n", "Numero sorteado é: ",num)

}

escreva ("; Quantidade de tentativas: ", tentativas)

}

}

4\_Jogo de Adivinhação

Implemente um jogo onde o computador sorteia um número entre 1 e 20, e o usuário tenta adivinhar. Após cada tentativa, informe se o número é maior ou menor que o chute. O jogo termina quando o usuário acerta.

programa {

inclua biblioteca Util --> u

funcao inicio() {

inteiro sorteado, num, tentativas = 0

sorteado = u.sorteia(1, 10)

escreva(sorteado)

enquanto ( num!= sorteado){

escreva("Digite o número que será sortido: ")

tentativas++

se (num==sorteado){

escreva ("\n", "Parabéns, o número sorteado é: ",sorteado, "\n")

} senao se (num<sorteado){

escreva ("O numero soreado é maior que o numero ",num, "\n")

} senao se (num>sorteado){

escreva ("O numero soreado é menor que o numero ",num, "\n")

}

}

escreva ("Quantidade de tentativas: ", tentativas)

}

}

5\_Classificador de Textos

Peça ao usuário que digite uma frase. Utilize a biblioteca Texto para converter toda a frase para caixa alta. Em seguida, exiba o número de caracteres da frase original.

programa {

inclua biblioteca Texto --> t

funcao inicio() {

cadeia palavra, capsLock

inteiro qtd

escreva("Escreva uma palavra: ")

leia(palavra)

qtd = t.numero\_caracteres(palavra)

capsLock = t.caixa\_alta(palavra)

escreva(capsLock)

escreva ("\n","A quantidade de letras na palavra '", palavra, "' é ", qtd)

}

}

6\_Verificador de Triângulo

Peça ao usuário três valores inteiros e verifique se eles podem formar um triângulo. Se sim, classifique-o como equilátero, isósceles ou escaleno usando estrutura SE...SENAO SE.

programa {

inclua biblioteca Matematica --> m

funcao inicio() {

inteiro lado1, lado2, lado3, resultMaior, soma = 0, isTriangulo=0

escreva("Escreva o um lado do triangulo: ")

leia (lado1)

escreva("Escreva outro lado do triangulo: ")

leia (lado2)

escreva("Escreva outro lado do triangulo: ")

leia (lado3)

resultMaior = m.maior\_numero(m.maior\_numero(lado1,lado2),lado3)

se (lado1 == resultMaior ){

soma = lado2 + lado3

se (soma > resultMaior){

isTriangulo = 1

} senao {

escreva ("Numeros não podem ser um triangulo1")

}

} senao se (lado2 == resultMaior ){

soma = lado1 + lado3

se (soma > resultMaior){

isTriangulo = 1

} senao {

escreva ("Numeros não podem ser um triangulo2")

}

} senao se (lado3 == resultMaior ){

soma = lado1 + lado2

se (soma > resultMaior){

isTriangulo = 1

} senao {

escreva ("Numeros não podem ser um triangulo3")

}

}

se (isTriangulo == 1){

se (lado1==lado2 e lado1==lado3 e lado2==lado3){

escreva("Este triangulo é equilatero")

} senao se (lado1!=lado2 e lado1!=lado3 e lado2!=lado3){

escreva("Este triangulo é escaleno")

} senao{

escreva("Este triangulo é isósceles")

}

}

}

}

7\_Média com Validação

Receba 3 notas de um aluno (entre 0 e 10). Se alguma nota for inválida, solicite novamente. Após isso, calcule e exiba a média, dizendo se o aluno está aprovado (>=7), em recuperação (>=5) ou reprovado.

programa {

inclua biblioteca Matematica --> m

funcao inicio() {

inteiro nota1, nota2, nota3

real media

faca{

escreva("Digite a nota 1: " )

leia (nota1)

se (nota1>10 ou nota1<0){

escreva("Valor incorreto, digite uma nota de 0 a 10","\n")

}

} enquanto(nota1>10 ou nota1<0)

faca{

escreva("Digite a nota 2: " )

leia (nota2)

se (nota2>10 ou nota2<0){

escreva("Valor incorreto, digite uma nota de 0 a 10","\n")

}

} enquanto(nota2>10 ou nota2<0)

faca{

escreva("Digite a nota 3: " )

leia (nota3)

se (nota3>10 ou nota3<0){

escreva("Valor incorreto, digite uma nota de 0 a 10","\n")

}

} enquanto(nota3>10 ou nota3<0)

media = (nota1+nota2+nota3)/3

escreva ("A média das notas ",nota1,",",nota2,",",nota3," é: ", media)

}

}

8\_Conversor de temperatura

Solicite ao usuário uma temperatura em graus Celsius e a converta para Fahrenheit. Peça ao final se ele deseja fazer outra conversão. Se sim, repita; se não, encerre o programa.

programa {

inclua biblioteca Matematica --> m

funcao inicio() {

real valor, convert

cadeia repete

faca{

escreva("Digite o valor em Celsius que será convertido em Fahrenheit ")

leia (valor)

convert = (valor \* 1.8) + 32

escreva("O valor de ",valor,"° Celsius é equivalente a ", convert,"° Fahrenheit")

faca{

escreva ("Deseja repetir?")

leia(repete)

}enquanto (repete!="sim" ou repete!="não")

} enquanto(repete=="sim")

}

}