

1 - declara variável texto chamada "texto"

leia texto e guarda na variável "texto"

declara variável inteira "quantidade"

quantidade = conta_caracteres(texto)

funcao conta_caracteres (texto_da_funcao) {

 declara variável "contador = 0"

 enquanto (texto_da_funcao[contador] != '/0'){

 contador = contador + 1

 }

 Retorne contador

 }

2- Declare variável "texto"

 Leia texto e guarde na variável "texto"

 Declare variável "resultado"

 resultado <- verifica_qtd_caracteres (texto)

 Função verifica_qtd_caracteres (texto) {

 Declare variável "retorno"

 Se o comprimento do "texto" > 0 então

 Retorno <- falso

 Se não

 Retorno <- verdadeiro

 Retorne retorno

 }

Fim se

Fim

3 –

Declare variável "s1"

Declare variável "s2"

Leia texto e guarde na variável "s1"

Leia texto e guarde na variável "s2"

Declare variável "resultado"

Se s1 tiver mais que 1 caracter então

 resultado <- verifica_cadeia (s1, s2)

Se não

 Mostre "s1 não tem caracteres"

Função verifica_cadeia (s1, s2) {

 Declare variável "retorno"

 Se s1 não está contido em s2 então

 Retorno <- -1

 Se não

 Retorno <- posição inicial de s1 em s2

 Retorne retorno

}

Fim se

Fim

4 –

declare variavel "texto"

leia do usuario e salve em "texto"

declare variavel textoContrario = ""

textoContrario = inverte_caracteres(texto)

imprime textoContrario

funcao inverte_caracteres (texto) {

 declare variavel texto_inverso = ""

 declara contador = texto.tamanho

```
    enquanto (contador >= 0) {  
        texto_inverso = texto_inverso + texto[contador]  
        contador = contador - 1  
    }  
    retorna texto_inverso  
}
```

5-

```
declare variavel "texto"  
leia do usuario e salve em "texto"  
declare variavel resultado  
resultado = testa_palindromo(texto)  
Mostre resultado  
se "texto" = "textoContrario"  
    imprime verdadeiro  
Senão  
    Imprime falso  
funcao testa_palindromo ( texto ) {  
    declare variavel texto_inverso = ""  
    declara contador = texto.length  
    enquanto (contador >= 0) {  
        texto_inverso = texto_inverso + texto[contador]  
        contador = contador - 1  
    }  
    Se texto = textoContrario  
        Retorna verdadeiro  
    Senão  
        Retorna falso  
}
```

6-

ler segundos

declara variável dias = segundos / 86400

declara x = 0

declara qtd_bisextos = 0

enquanto (x > 0):

se (VerificaSeBisexto(1970+x))

qtd_bisextos = qtd_bisextos + 1

x = x + 1

declara dias_ajustado = dias - qtd_bisextos

declara data_formatada = CalculaDataFormatada(dias_ajustado)

imprime data_formatada

funcao CalculaDataFormatada (dias_ajustado) {

declara variavel anos = inteiro(dias_ajustado / 365)

declara anos_ajustado = anos + 1970

declara dias_sobra = dias_ajustado - (anos * 365)

declara texto qtd_meses = "01"

declara inteiro dias_restantes = 1

Se (dias_sobra > 31) { qtd_meses = "02"; qtdMeses = 2 }

Se (dias_sobra > 60) { qtd_meses = "03"; qtdMeses = 3 }

Se (dias_sobra > 91) { qtd_meses = "04"; qtdMeses = 4 }

Se (dias_sobra > 121) { qtd_meses = "05"; qtdMeses = 5 }

Se (dias_sobra > 152) { qtd_meses = "06"; qtdMeses = 6 }

Se (dias_sobra > 182) { qtd_meses = "07"; qtdMeses = 7 }

Se (dias_sobra > 213) { qtd_meses = "08"; qtdMeses = 8 }

Se (dias_sobra > 244) { qtd_meses = "09"; qtdMeses = 9 }

Se (dias_sobra > 274) { qtd_meses = "10"; qtdMeses = 10 }

```

Se (dias_sobra > 305) { qtd_meses = "11"; qtdMeses = 11 }
Se (dias_sobra > 335) { qtd_meses = "12"; qtdMeses = 12 }
declara dias_restantes = dias_sobra - (qtdMeses * 30)
retorna "dias_restantes/qtd_meses/anos"
}

```

```

funcao VerificaSeBisexto ( ano ) {
    Declare ano Inteiro;
    Declare bissexto Booleano;
    Se ( ano módulo 400 é 0 ) então
        bissexto = Verdade
    Senão
        Se (ano módulo 4 é 0 E ano módulo 100 é diferente de 0) então
            bissexto=Verdade
        Senão
            bissexto=Falso
    retorna bissexto
}

```

7-

```

declara valor_fahrenheit
ler quantidade de valor_fahrenheit
declara valor_centigrados = (valor_fahrenheit - 32) / 1.8
imprime valor_centigrados

```

8-

```

ler vetor_moedas
ler vetor_notas
declara total_dinheiro = ContaMoedas(vetor_moedas) + ContaNotas(vetor_notas) |
196 + 600 = 796 centavos
total_notas = inteiro(total_dinheiro / 100)

```

```
total_centavos = total_dinheiro modulo 100
```

```
imprime "R$ total_notas e total_centavos centavos"
```

```
funcao ContaMoedas (vetor_moedas){
```

```
    declaro soma_moedas = 0
```

```
    declaro vetor moeda[4] = {1, 5, 10, 25, 50}
```

```
    faca(inteiro i=0; i < vetor_moedas.tamanho; i++){
```

```
        soma_moedas = soma_moedas + (moeda[i] * vetor_moedas[i])
```

```
    }
```

```
    retorna soma_moedas
```

```
}
```

```
funcao ContaNotas (vetor_notas){
```

```
    declaro soma_notas = 0
```

```
    declaro vetor notas[6] = {2, 5, 10, 20, 50, 100, 200}
```

```
    faca(inteiro i=0; i < vetor_notas.tamanho; i++){
```

```
        soma_notas = soma_notas + (notas[i] * vetor_notas[i])
```

```
    }
```

```
    soma_notas = soma_notas * 100
```

```
    retorna soma_notas
```

```
}
```

9-

ler segundos

declara variável dias = segundos / 86400

declara variavel anos = inteiro(dias / 365)

declara diferenca = dias - (anos * 365)

declara meses = arredonda pra cima (diferenca / 30)

declara minutos = (segundos modulo 86400) * 60

imprime

10-

ler x1

ler x2

ler y1

ler y2

declara resultado = calcula(x1, x2, y1, y2)

imprime resultado

função calcula(x1, x2, y1, y2){

declara parcial_1 = (x2 - x1)(x2 - x1)

declara parcial_2 = (y2 - y1)(y2 - y1)

declara resultado = raiz_quadrada(parcial_1 + parcial_2)

retorna resultado

}