
MANUAL DE INSTRUÇÃO

TRANSPILADOR

Anhanguera 2024

INTRODUÇÃO

O transpilador transcreve um Código escrito em Português Estruturado para uma versão equivalente em Linguagem C.

Nosso transpilador foi desenvolvido para uso didático, portanto não será 100% preciso em todas as correções e funções que um compilador para uso profissional contém. Todavia, seguindo as instruções que serão apresentadas neste manual, a qualquer falha no seguimento destas ele indicará qual o possível erro cometido.

LIMITAÇÕES

NOSSO TRANSPILADOR CONTÉM ALGUMAS RESTRIÇÕES AO CODIFICAR:

-Não permite utilização de comentários.

Ex: // "comentário"

-Não aceita a declaração de funções.

-Todo o Código digitado no arquivo .txt será **automaticamente gerado** dentro de uma função **"int main()"**. Portanto, **não é necessário/permitido declarar em português estruturado a "função inicio" e "final"**.

-Não aceita a declaração de bibliotecas. As bibliotecas são geradas automaticamente após a conversão do código.

DECLARAÇÃO DE VARIÁVEIS

Portugol	Linguagem C
inteiro	int
decimal	float
real	double
logico	bool
caractere	char
literal	char com 100 caracteres
para	for
enquanto	while
escreva	printf
leia	scanf
verdadeiro/falso	true/false

Portugol	Linguagem C
se	if
senao	else
senaose	else if
igual	==
diferente	!=
maior	>
menor	<
ou	
ee	&&
menorigual	<=
maiorigual	>=
<-	=

MODELO DE UMA DECLARAÇÃO DE VARIÁVEIS

Tipo da variável	Nome da variável	Atribuição	Valor atribuído
inteiro	Nome	<-	Número inteiro (Ex: 10)
real	Nome	<-	Número Decimal(double) (Ex: 0.5)
logico	Nome	<-	Booleano (Ex: verdadeiro/falso)
decimal	Nome	<-	Número Decimal (Ex: 2.5)
caractere	Nome	<-	Caractere Único (Ex: 'a')
literal	Nome	<-	Multiplos Caracteres (Ex: "Maria")

MODELO DO “ESCREVA” (PRINTF)

```
escreva ("Olá, mundo")|
```

MODELO DO “LEIA” (SCANF)

```
leia (%d , NomeDaVariavel)
```

MODELO DO “PARA” (FOR)

Obs: A variável contador deve ser declarada antes da estrutura “para”.

```
para (s <- 0; s maior 10; s++){  
    /*conteúdo de código*/  
}
```

MODELO DO “ENQUANTO” (WHILE)

```
enquanto (Condição1 diferente condição2)  
{  
    escreva("--Exemplo--")  
}
```

MODELO DO “SE / SENAPOSE / SENAO” (IF/ ELSE IF / ELSE)

```
se (chute menor numero_secreto)
{
  escreva("O número secreto é maior.")
}
senapose (chute maior numero_secreto)
{
  escreva("O número secreto é menor.")
}
senao
{
  escreva("Parabéns! Você acertou o número secreto")
}
```

EXEMPLO DE UM CÓDIGO FUNCIONAL

```
inteiro numero_secreto, chute, tentativas
numero_secreto <- 7
tentativas <- 0

escreva("Bem-vindo ao Jogo de Adivinhação!")
escreva("Tente adivinhar o número secreto (entre 1 e 10).")

enquanto (chute diferente numero_secreto)
{
  escreva("Digite seu palpite: ")
  leia(%d, chute)
  tentativas <- tentativas + 1

  se (chute menor numero_secreto)
  {
    escreva("O número secreto é maior.")
  }
  senapose (chute maior numero_secreto)
  {
    escreva("O número secreto é menor.")
  }
  senao
  {
    escreva("Parabéns! Você acertou o número secreto em %d tentativa(s)!", tentativas)
  }
}
```