

MOISES PEDRO MARCANTE MONTEIRO ----- TURMA B

1 - Faça a representação de um número decimal, Hexadecimal e Binário.

Hexa: $5F6 = 5 \times 16^2 + 15 \times 16^1 + 6 \times 16^0$

Decimal: $521 = 5 \times 10^2 + 2 \times 10^1 + 1 \times 10^0$

Binario: $100101 = 1 \times 2^5 + 0 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$

2 - O que é uma constante?

Uma constante é um elemento que pode guardar um dado **constantemente**, isto é, durante todo o processo de execução do algoritmo, o valor desse elemento não irá sofrer alteração .

3 - Quando se pode usar uma variável?

Em qualquer caso que seja necessário, como por exemplo em uma entrada de usuário para inserir um valor que vai ser modificado durante a execução.

4 - O que é um identificador? Dê exemplo

R: Um identificador é um nome, ou número, ou uma junção de ambos, que é usado para identificar VARIÁVEIS e CONSTANTES, por exemplo.

5 - Dê ao menos dez exemplos de tipos de dados (Primitivos).

R: Inteiro;Flutuante;Booleano;Caractér;Data;JSON;Time;Long Int;Varchar;Decimal;

6 - Operador Aritmético

$+, -$	Operadores unários, isto é, são aplicados a um único operando. São os operadores aritméticos de maior precedência. Exemplos: -3 , $+x$. Enquanto o operador unário $-$ inverte o sinal do seu operando, o operador $+$ não altera o valor em nada o seu valor.
\backslash	Operador de divisão inteira. Por exemplo, $5 \backslash 2 = 2$. Tem a mesma precedência do operador de divisão tradicional.
$+, -, *, /$	Operadores aritméticos tradicionais de adição, subtração, multiplicação e divisão. Por convenção, $*$ e $/$ têm precedência sobre $+$ e $-$. Para modificar a ordem de avaliação das operações, é necessário usar parênteses como em qualquer expressão aritmética.
MOD ou %	Operador de módulo (isto é, resto da divisão inteira). Por exemplo, $8 \text{ MOD } 3 = 2$. Tem a mesma precedência do operador de divisão tradicional.
$^$	Operador de potenciação. Por exemplo, $5^2 = 25$. Tem a maior precedência entre os operadores aritméticos binários (aqueles que têm dois operandos).

7 - Operadores Relacionais

= Significa IGUAL A

< Significa Menor que

> Significa Maior que

=> Significa maior ou igual que

<= Significa menor ou igual que

```

algoritmo
  declare a, b, j numerico
  a <- 1
  b <- 100
  para j <- a ate b faca passo 1
  inicio
    a <- 2
    b <- 8
    escreva j
  fim
  escreva a, " ", b
fim_algoritmo

```

8- a, b e j são variáveis declarados como numérico

a recebe o valor 1

b recebe o valor 100

No laço PARA j recebe o valor de a até chegar no valor de b a cada execução, ele faz o passo 1

No passo 1 a recebe 2 e b recebe 8 e no fim ele escreve o valor de j

Fora do laço, o algoritmo escreve o valor de a, um espaço , e o valor de b.

9 - Qual a diferença entre linguagens compilada e interpretada?

As linguagens interpretadas são aquelas que o sistema usa sem passar para o nível de máquina (binário).

As linguagens compiladas são aquelas que passam por um processo de compilação, tornando-se em binário, executando o algoritmo, e retornando o processo em um console por exemplo.

10 - O que é linguagem, Case – sensitive?

São linguagens com regras estritas em relação a nomenclatura de variáveis e outros elementos que podem ser nomeados. Por exemplo NOME é diferente de nome.

11 - Complete a tabela abaixo:

Código	Significado
\n pula linha	
\t pula espaço na mesma linha	
\\ insere uma barra	
\0 termina uma string	
\' insere um apóstrofo dentro do print	
\” insere aspas	
\a retorna um som, bip, alarme	

12 - Comente o código, linha por linha:

```

#include "stdafx.h" //inclui a biblioteca
int
main() {
DECLARA A
MAIN
    int numero = 50
DECLARA A VARIÁVEL NUMERO
COMO INTEIRO

float valor = 8.37; DECLARA

```

A VARIÁVEL VALOR COMO
FLUTUANTE

`char` letra = 'X'; DECLARA A
VARIÁVEL LETRA COMO CARACTER

`int` valor1, valor2, valor3; DECLARA EM SEQUÊNCIA A VARIÁVEL VALOR COMO INTEIROS

```
valor1 = valor2 = valor3 = 20;          VALOR1, 2 E 3 RECEBEM VALOR INTEIRO 20
printf("Variável número...:%d          \n", numero);
printf("variável valor...:%f           \n", valor);
printf("Variável valor...:%10.2f       \n", valor);
printf("Variável valor...:%.2f         \n", valor);
printf("Variável letra...:%c           \n", letra);

printf("%d %d %d          \n", valor1, valor2, valor3);

return                                0;

}
```

ESCREVE O VALOR DA VARIÁVEL NÚMERO
ESCREVE O VALOR DA VARIÁVEL VALOR FLUTUANTE
ESCREVE O VALOR DA VARIÁVEL VALOR PORCENTO
ESCREVE O VALOR DA VARIÁVEL VALOR ARRENDONDADO PARA DUAS CASAS
ESCREVE O VALOR DA VARIÁVEL LETRA
ESCREVE O VALOR DAS VARIÁVEIS VALOR1, 2 E 3

13 - Linguagem C

Escreva um programa que declare variáveis. Estas variáveis irão armazenar:

```
#include <stdio.h>
#include <stdlib.h>
```

```
Int main(){
```

```
int Km_inicial = 200000;
int Km_final = km_inicial + 5600; //Viagem de 5600 quilômetros percorridos
```

```
Quilometragem = km_final – km_inicial;
```

```
Printf("a quilometragem depois da viagem é de : %d ",quilometragem");
```

```
Return 0 ;}
```