Trabalho de pesquisa

***1. Tipos de variáveis:***

* Integer (para números inteiros);
* Real (para números fracionários);
* Boolean (para variáveis booleanas –“**True”** ou “**False”**);
* Char (para representar apenas um dígito ou uma letra);
* String (conjunto de uma ou mais letras e/ou dígitos).

***2. Operadores aritméticos e lógicos:***

**Aritméticos:**

* **“+,-“ 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.
* **“\” Operador de divisão inteira.** **-** Por exemplo, 5 \ 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 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).

**Lógicos:**

* **“NOT” Operador unário de negação.** **-** not VERDADEIRO = FALSO, e not FALSO = VERDADEIRO. Tem a maior precedência entre os operadores lógicos.
* **“OR” -** Operador que resulta VERDADEIRO quando um dos seus operandos lógicos for verdadeiro.
* **“AND” -** Operador que resulta VERDADEIRO somente se seus dois operandos lógicos forem verdadeiros.
* **“XOR” -** Operador que resulta VERDADEIRO se seus dois operandos lógicos forem diferentes, e FALSO se forem iguais.

***3. Comandos de Leitura e Escrita:***

* **“Read** e **Readln” –** Lê o código para que apareça no ecrã; ambos desempenham a mesma função, a unica diferença é que após a entrada de dados com **Read** o cursor fica na mesma linha, e no caso do **Readln** o cursor vai para a linha de baixo.
* **“Writeln” -** mostra o seu conteúdo e passa o cursor para a linha de baixo.
* **“Write” -** mostra o seu conteúdo e mantém o cursor na mesma linha.

***4. Estruturas de decisão (IF e CASE):***

* **“If” -** A palavra-chave **if** precede uma condição, deve ser seguida por **then** e por uma instrução. A declaração pode opcionalmente ser seguida por **else** e outra declaração.

Se você precisar de duas ou mais instruções, coloque-as dentro de um **begin**… **end**.

* **“Case” -** O comando **CASE** é importante para a estruturação de um programa que possua diversas opções de execução, tornando-o bem legivel e estruturado, evitando o uso do **IF** de forma repetida.A palavra-chave **case** precede um **seletor** e deve ser seguida por **of.**