Manual da linguagem Asdf

Variáveis:

A declaração de variáveis pode ser feita em qualquer lugar do código menos dentro de um loop. Podem ser declaradas de forma separada ou conjunta(se todas forem do mesmo tipo).

Ex:
TIPO var1, var2, var3;

Pode ocorrer atribuições de valores logo na criação:

Ex:
TIPO var1 = 1, var2 = 2, var3 = 3;

Existem três tipos de variáveis são eles: inteiro, real e str.
O tipo inteiro aceita apenas números inteiros, o tipo real aceita tanto números inteiros como números racionais(não podem ser feitas operações aritméticas entre os dois tipos), e o tipo str aceita textos.

A atribuição de textos a variável do tipo str deve ser feita entre " ".
Ex:
str text = "asdf";

Se inicializadas sem nenhum valor variáveis do tipo str receberam um texto vazio, variáveis para números receberam 0.

Operações:

As operações aritméticas devem ser feitas apenas com dois operandos por vez, e com variáveis do mesmo tipo.
Ex:
inteiro a, b;

a = 10, b = 4;
a = a + b;

O processo é análogo para variáveis do tipo real, variáveis do tipo str podem ser alteradas diretamente como: str text =
"abcde";

text = "aaaa";

operadores: *, +, -, /, %.

Comparações:

As comparações podem ser feitas apenas entre duas variáveis.
Ex:
inteiro a = 20, b;

se(b < a){

}

O comando se
será explicado
logo a seguir.

Operadores de comparação: <, <=, ==, !=, >=, >

OBS: Comparações do tipo a < 10, b == 5 ou seja entre número e variável geraram um erro de sintaxe.

Comando se:

O comando se aceita uma comparação para a decisão de executar ou não o bloco de código que é definido por '{'(inicio) e '}'(fim).

Ex:

```
inteiro a = 2, b = 3;

se( a != b ){
        a = 0;
}
```

OBS: Comandos na mesma linha que o caracter { serão ignorados.

Laço de repetição:

O comando enquanto aceita uma comparação e executa o bloco seu bloco de código até que a condição seja verdadeira.

Ex:

```
inteiro i = 0, cmp = 10;

enquanto( i < cmp ){
        i = i + 1;
}
```

Entrada e Saída:

Para realizar a entrada de dados pode-se utilizar do comando entrada.

O comando entrada aceita um número indeterminado de variáveis e tenta ler da entrada dados que correspondem aos tipos das variáveis ex:

```
inteiro a;
real b;
str c;

entrada( a, b, c );
```

O comando irá primeiramente encontrar um inteiro para a variável a e depois um real para a variável b e finalmente um texto para a variável c;

Obs: Quando se trata de variáveis do tipo str qualquer coisa pode ser um texto.

Para realizar a saída de dados pode-se utilizar o comando saida.

O comando saída também aceita um número indeterminado de variáveis ou de textos(não deve estar entre aspas).

Ex:

inteiro p:
saida(Verificaremos se, p, eh primo);

Todos os dados devem ser separados por ','.

Nenhuma indentação ou espaçamento é requerida na sintaxe da linguagem, só atentamos para o fato de que comandos nas linhas onde existe uma palavra chave ou de tokens como '}' vão ser ignorados.

Ex:
se(a == b){ estou sendo ignorado.

} vou ser ignorado.

Esses lugares podem ser utilizados como linha de comentário.