Sintaxe aceita pelo compilador:

O compilador original era case sensitive e só permitia as palavras reservadas minúsculas.

Introduzimos um comando para converter na hora da leitura da palavra para minúsculo, logo agora o compilador é case insensitive.

Segue abaixo um resumo da sintaxe, isso é só pra facilitar os testes do compilador e posteriormente iremos escrever toda a sintaxe pela grámatica EBNF para introduzir no trabalho escrito.

Tipos de variáveis:

caracter;

inteiro;

real;

logico;

literal;

OBS: O tipo literal foi implementado, mas ainda estamos implementando as operações com strings, até o momento o que está funcional são as comparações.

Estrutura principal:

programa <nome>();

<declarações>

inicio

.<instruções>

.<instruções>

fim.

Declaração de constantes:

const

<nome-constante> = <valor1>;

.

.

.

<nome-constante> = <valorn>;

Declaração de tipos:

tipos

<nome-tipo1> = <tipo1>;

.

.

.

<nome-tipon> = <tipon>;

Declaração de variáveis:

var

<nome-variavel1> : <tipo1>;

.

.

.

<nome-variaveln>:<tipon>;

Procedimentos:

procedimento <nome>(<parametro1> ... <parametron>);

.<declarações>

inicio

.<instrução>

.<instrução>

fim;

Funções:

funcao <nome>(<parametro1>...<parametron>):<tipo-retorno>;

<declarações>

inicio

.<instruções>

.<instruções>

fim;

OBS1: Para a passagem de parametro por referência, utiliza a palavra 'ref' na declaração dos parametros da função.

OBS2: O compilador não comporta foward e procedimentos e funções, então todo o corpo da função e suas instruções devem ser escritos antes de iniciar as instruções do programa.

Estruturas de repetição:

para <variavel> de <valor-inicial> ate <valor-final> faca

inicio

.<instrução>

.<instrução>

.<instrução>

fim;

OBS: O compilador verifica o valor inicial e final, não havendo necessidade de entrar com a palavra reservada "to" ou "downto", caso o usuário entre com um valor inicial maior que o final ele reconhece como "downto".

repita

.<instrução>

.<instrução>

.<instrução>

ate (<condição>);

enquanto (<condição>) faca

.<instrução>

.<instrução>

.<instrução>

fim;

Estruturas de decisão:

caso (<variavel>) de

<label1> : <instrução1>;

.

.

.

<labeln>:<instruçãon>;

fim;

OBS1: O compilador ainda não suporta uma label default na estrutura caso.

OBS2: É possível introduzir mais de uma label por instrução.

se (<condição>) entao

inicio

.<instrução>

.<instrução>

.<instrução>

{senao

inicio

.<instrução>

.<instrução>

.<instrução>

}

fim;

O comando entre {} é opcional.

Entrada/Saída de dados:

escreva('<string-literal>'); ou escreva(<nome-variavel>);

OBS: <string-literal> e <nome-variavel> podem ser escritas no mesmo procedimento utilizando ',' como separador.

OBS2: Para a quebra de linha, utilize '\n'.

leia(<nome-variavel);