

Great Sage

Uma Nova Linguagem de Programação

Leonardo Pereira Medeiros

Definição

Great Sage é uma linguagem de programação baseada em C, com alguns componentes de VBA.

Essa Linguagem utiliza de flags e operadores em siglas para tornar mais fácil e intuitivo o processo de programar.

The logo for Great Sage is displayed on a dark background. It consists of the words "GREAT" and "SAGE" in a bold, sans-serif font. "GREAT" is in a light blue color, and "SAGE" is in a light yellow color. The text is contained within a dark rectangular box with a thin white border.

GREAT SAGE

Estrutura(declaração)

- Great sage é uma linguagem de tipagem forte, que utiliza os tipos Integer e Boolean.
- Usa-se flags para declarações de variáveis e funções. Deixando-as mais evidentes e modulares.

```
--Função com retorno--
```

```
__function__ type funcName(args){  
    statements  
}
```

```
--Função sem retorno--
```

```
__sub__ subFuncName(args){  
    statements  
}
```

```
--Variável--
```

```
__var__ type varName
```

Estrutura(operações)

- Os Operadores são siglas, a fim de evitar o uso de símbolos, para que o processo de escrita seja mais fácil, independente do teclado utilizado.

OR	and
AND	or
NOT	not

ADD	sum
SUB	subtraction
MUL	multiplication
DIV	division

=	equal (comparação e associação)
LT	less than
GT	greater than

Estrutura(condições)

- Os tokens `if` e `else` representam os comandos de condição.

--Estrutura--

```
if (condition){  
    statements  
}
```

```
else{  
    statements  
}
```

--Exemplo--

```
if (a GT b){  
    c = True  
}
```

```
else{  
    c = False  
}
```

Estrutura(loop)

- 0 token while representa o comando de loop.

--Estrutura--

```
while(condition){  
    statements  
}
```

--Exemplo--

```
while(a LT b){  
    a = a ADD 1  
}
```

Estrutura(chamadas)

Há dois tipos de chamada de função:

- Com a Flag Call: para funções sem retorno(__sub__).
- Sem a Flag Call: para funções com retorno(__function__).

```
-- SEM Retorno --
```

```
CALL subFunctionName(args)
```

```
-- Com Retorno --
```

```
Var = FunctionName(args)
```

Estrutura(outros componentes)

- Comentários no código devem ser indicados com o token '.
- Para printar um elemento usa-se o comando print.
- O token input representa o comando que recebe um valor de entrada do usuário.

--Exemplo--

```
' hello Word
__sub__ main(){
    __var__ integer a
    a = input
    print a
}
```


Demonstração

- Este é exemplo simples de um código que calcula fatorial na estrutura do Great Sage.
- Obs: a execução do programa sempre começa pela função main.

```
__sub__ numbers(integer n){
    while(n GT 0){
        print n
        n = n SUB 1
    }
}

__function__ integer Fatorial(integer n){
    if (n = 0){
        Fatorial = 1
    }
    else{
        Fatorial = n MUL Fatorial(n SUB 1)
    }
}

__sub__ main(){
    __var__ integer n
    __var__ boolean flag
    n = input
    flag = True
    if (flag){
        CALL numbers(n) 'componentes
    }
    print Fatorial(n) 'resultado do fatorial
}
```