



UNIVERSIDADE FEDERAL DA FRONTEIRA SUL - UFFS

ACADÊMICA: RAFAELLE LARISA DE ARRUDA

## Interpretador em Java Manual do Rafis Chuchu

Chapecó – SC

2019/01

## Apresentação da Linguagem

A linguagem Rafis Chuchu fornece uma sintaxe flexível, imperativa, orientada a objetos, funcional, e de auto nível. Criada pela @mazarafa, entretanto, possui uma tripagem dinâmica, e a sua principal característica é permitir a fácil leitura do código, e exige poucas linhas de código se comparando com outras linguagens.

## **INTRODUÇÃO**

É de todo conhecimento geral que, este trabalho foi desempenhado para interpretar uma linguagem dinâmica, escrita com poucas linhas de código. Rafis é uma linguagem de programação de altíssimo nível, criada pela @mazarafa sobre a ideia de “Programação de computadores para todos”, com esse desenvolvimento, teve em mente a liberdade, código livre, tendo flexibilidade de ser interpretado.

## EXPRESSÕES DO RAFIS CHUCHU

Aritméticas:

**+** : soma  
**-** : subtração  
**\*** : multiplicação  
**/** : divisão  
**%** : resto

Ressaltando que na expressão, podemos ter no mínimo 2 operandos e são sequências. Ex.:  $2+2*3=12$ . **OBS: Começa com o primeiro operador, soma e depois multiplica.**

Na propriedade lógica dos sinais:

**!** : desigualdade  
**<** : menor  
**>** : maior  
**&** : igualdade

Essas expressões lógicas, podem contém obrigatoriamente dois operandos. A expressão não aceita booleanos.

### Sintaxe da Linguagem

A sintaxe refere-se às regras que regem a composição dos textos em uma linguagem formal que constitui de um sistema lógico. Podendo fornecer uma interpretação. Na ciência da computação, a sintaxe diz às regras que regem a composição de textos com significado em uma linguagem formal, tal como uma linguagem de programação, fazendo sentido definir a semântica ou significado, ou fornecer uma interpretação.

var - Declaração de Variáveis

**Sintaxe: var**

- Variável recebe o valor 0.

**Sintaxe: var nome\_da\_variável = 0;**

- Variável recebe o valor 5.0.

**Sintaxe: var nome\_da\_variável = a + 5.0;**

- Variável recebe o resultado da expressão.

if - Controlador de fluxo

Quando programamos, muitas vezes precisamos que determinado bloco de código seja executado apenas se uma determinada condição for verdadeira. Em casos assim, devemos fazer uso de uma estrutura de condição.

**Exemplo:**

**Sintaxe: if condição**

- Se a expressão for VERDADEIRA executa as instruções até o final do controle de fluxo;
- O fim de controle de fluxo é representado por **end if**;

while – Laço

O while repete um código de instrução enquanto a condição definida em seu cabeçalho for verdadeira. A estrutura while talvez seja a mais simples para entendermos nesse momento, porém, não raramente encontramos alunos que dizem não entender o funcionamento dessa estrutura. Se você entendeu o funcionamento da estrutura if, pense na estrutura while como sendo a estrutura if, mas que ao invés de executar o seu código de instrução uma única vez, executará enquanto a expressão definida for igual a verdadeira.

**Exemplo:**

**Sintaxe: while condição**

- Se for VERDADEIRA executa as instruções até o final do laço ou até algum comando de desvio incondicional;
- O comando **'break'** é um desvio incondicional. Ele quebra o laço que esta sendo executado.
- Com o fim do laço, representamos com o comando **end while**.

get - Comando de Entrada

**sintaxe : get nome\_da\_variável**

- get recebe um valor que tem como origem do teclado. Uma variável com esse nome será criada e atribuída um valor. Ex: é como se fosse um "print" ou "println".

show - Comando de saída

Show sempre pula uma linha no final de sua execução.

**sintaxe: show 'string'**

- Imprime "string" na tela.

**sintaxe: show 'string' > 'string2'**

- É possível concatenar strings através do comando '>' para imprimir na tela. O mesmo vale para variáveis ou expressões matemáticas.

## EXEMPLOS DE IMPLEMENTAÇÃO DO RAFIS CHUCHU.

I) O código a seguir imprime a média entre dois números.

```
get x
get y
show x + y / 2
```

II) O código a seguir imprime “é primo” se um número maior ou igual a 2 for primo. Se não, ele imprime “não é primo”;

```
get x
var y = 2
var bool
while y < x
- - - - - if x%y & 0
- - - - - show ' não é primo'
- - - - - var bool = 1
- - - - - break
- - - - - end if
- - - - - var y = y + 1
end while
if bool & 0
- - - - show 'é primo'
end if
```

III) O código imprime a tabuada do 1 ao 10;

```
var x
var y = 1
while y < 11
- - - - var x=1
- - - - show 'tabuada do ' > y
- - - - show ' '
- - - - while x < 11
- - - - - show x > ' * ' > y > ' = ' > x*y
- - - - - var x = x + 1
- - - - end while
- - - - var y = y + 1
- - - - show ' '
end while
```

IV) Imprimido na tela frase ou números inteiros, tanto como números flutuantes.

```
show 'Hello world Rafis Chuchu'
show ' Vamos pro Tomorrowland belgium parça !'
show ' 1 + 1 // tela 2
```

V) Aplicação com os operadores:

```
var a = 1 + 1  
var b = 2 * 2  
var c = 2 + 2 * 2
```

```
show a  
show b  
show c
```

Observação: A linguagem não aceita métodos do tipo “()” ou “[ ]” e “{}” no cálculo. Lembre-se que o primeiro operador que fez a conta.

**Ex = 2 + 2 \* 2 = 8**

IV) Caso deseje comentar o código, use sempre o “#”. Esse método funciona apenas para cada linha que deseja comentar.

```
#show 'Teste'
```

## Execução da Linguagem Rafis Chuchu

Para executar seu código, digite na linha de comando:

```
java -jar Rafis.jar ./diretório/do/arquivo/nome_do_arquivo.rafis
```

Exemplos de como executar o programa.

