Funções matemáticas (Raizes)

Para definir uma raiz quadrada a melhor forma seria através de uma função **sqrt()**, pois seria mais facilmente associada a outras linguagens que recorram a bibliotecas matemáticas com esta mesma notação.

Para a raíz quadrada teríamos então:

**Sqrt(**parâmetro **)**

Onde o parâmetro seria único e um valor real ou inteiro positivo.

Em javascript basta utilizar **Math.sqrt()** para a raiz

Para as **n raízes** (cúbica, à quarta, etc..) será necessário criar uma função custom **sqrt()** que recebe-se no máximo 2 parâmetros onde o primeiro seria o valor a calcular e o segundo o valor do radical, onde poderíamos recorrem ao **math.pow** determinar as n raizes. Caso não seja indicado nenhum valor no radical seria utilizada a função acima, com o primeiro parâmetro na raiz.

**Sqrt(** param1**,**param2 **)**

Uma das soluções possíveis em javascript, que suportam negativos:

function sqrt(x, n) {

//caso a radical não seja definido faz a raiz quadrada

if(typeof n === 'undefined'){

return Math.sqrt(x);

}else{

//valor do radical ímpar e raiz menor que 0

var negativo = n % 2 == 1 && x < 0;

if(negativo)

x = -x;

var valor = Math.pow(x, 1 / n);

n = Math.pow(valor, n);

if(Math.abs(x - n) < 1 && (x > 0 == n > 0))

return negativo ? -valor : valor;

}

}

**Testes:**

Resultado esperado:

5

parser.parse("sqrt('25','2')").evaluate();

Ou

parser.parse("sqrt('25')").evaluate();

Resultado esperado:

3

parser.parse("sqrt('27','3')").evaluate();