

# Primeiro Trabalho de Estruturas de Linguagens

Professor: Tiago A. O. Alves e-mail: tiago@ime.uerj.br

June 2, 2015

O objetivo deste trabalho é implementar (em linguagem C) um interpretador para uma linguagem **extremamente** simples. Daremos à essa linguagem o nome de Soma++. As únicas operações que existem em Soma++ são as diferentes amarrações feitas ao operador + e os únicos tipos de dado existentes em Soma++ são strings de caracteres, números inteiros e números de ponto flutuante. Por exemplo, uma operação válida em Soma++ é: `1 + 1`. E a resposta que o interpretador deverá dar para essa linha de entrada é, obviamente, o número 2.

Além das operações de soma, Soma++ também permite que o operador + seja utilizado para concatenar strings, por exemplo:

```
"Universidade do " + "Estado do Rio de Janeiro"
```

deverá retornar a string

```
"Universidade do Estado do Rio de Janeiro"
```

Todos os tipos da linguagem Soma++ são compatíveis entre si, obedecendo a seguinte regra de coerção de tipos:

1. Caso um dos operandos seja String, o outro deve ser convertido para String, caso não seja. A operação exercida pelo operador +, então, é a concatenação de strings.
2. Caso um dos operandos seja ponto flutuante e o outro inteiro, o operando inteiro deve ser convertido para ponto flutuante.

Dessa forma, os seguintes exemplos de entrada/saída são válidos:

```
"Maio " + 19 + ", " + 2015
```

```
"Maio 19, 2015"
```

```
12.0 + 1
```

```
13.0
```

```
"Maria pesa " + 60.2 + " kg"
```

```
"Maria pesa 60.2 kg"
```

As operações devem ser sempre avaliadas da esquerda para a direita, a não ser que sejam usados parênteses para priorizar operações. Por exemplo:

```
"Dia " + 18 + 1 " de Maio"
```

Deverá gerar a saída

```
"Dia 181 de Maio"
```

Ao passo em que:

```
"Dia " + (18 + 1) + " de Maio"
```

Deverá gerar a saída:

```
"Dia 19 de Maio"
```

Bônus (valendo 1 ponto a mais): Implemente variáveis que poderão guardar os valores de uma linha de operações de Soma++. Quando se atribui um valor a uma variável, o valor não será escrito na saída padrão. Por exemplo:

```
x = 10  
y = x + 9  
"Dia " + y + " de Maio"
```

Gerará apenas a saída:

```
"Dia 19 de Maio"
```