

Enunciado Laboratório 03 de Compiladores.
Entrega 13/11/2021

Você deve:

- Fazer o trabalho individualmente;
- Implementar e enviar um programa com a solução;
 - Sua solução deve ser em Java e utilizar o esqueleto disponibilizado no .zip;
- Gravar e enviar um vídeo de até 15min (esperado 10min ou menos) explicando o seu programa;
 - Sua explicação deve conter:
 - Visão geral da solução
 - Como você checkou a tipagem dos operandos de operadores

Realizar os envios **SOMENTE** pelo Classroom, até 13/11/2021.

Complete o programa em Java do .zip disponibilizado para executar uma versão reduzida de checagem de tipos de uma função.

A função a ser preenchida é a função **correctTypes** do arquivo **CheckType.java**. Esta função recebe um objeto do tipo **FunctionDeclaration**. Este objeto recebido é uma declaração de uma função. Esta declaração de função estará mais restrita do que o caso geral; todos os tipos de objetos que ocorrerem nela serão somente um dentre **int** e **boolean**. Além disso, a declaração de função passada não conterá expressões dos tipos i) chamadas de função, ii) **length** de objeto, iii) **this** ou iv) **new**.

Seu objetivo é decidir, retornando verdadeiro ou falso, se a tipagem de todos os comandos (**Statement**) e expressões (**Expression**) da função estão corretos. Você deve escrever o código de **correctTypes** para isso. Sua checagem deve verificar se os tipos dos parâmetros de operadores estejam corretos e se seus retornos estão coerentes e se as atribuições estão recebendo valores do mesmo tipo, etc.

Por exemplo, o código:

```
boolean a;
```

```
int b;
```

```
int c;
```

```
a = b < c;
```

está correto, pois o retorno de uma comparação **<** é um boolean e os parâmetros de uma comparação **<** são inteiros. Já o código

```
boolean a;
```

```
int b;
```

```
int c;
```

```
c = (b + c) < (b - c) ;
```

está errado, pois o lado direito da atribuição tem valor booleano. Já o código

```
boolean a;
```

```
int b;
```

```
int c;
```

a = (b + c) < (b < c) ;

está errado, pois o segundo parâmetro da comparação < tem tipo booleano.

A gramática utilizada é a mesma do lab 2. A saída do lab 2 pode ser usada como entrada para este lab. No entanto, isto não é necessário. Objetos do tipo **FunctionDeclaration** podem ser passados diretamente para a função **correctTypes** do objeto **checkType**.