

- (1) A implementação atual do compilador didático aceita apenas números com um único dígito. Modificar o compilador para que ele aceite também números naturais com mais de um dígito
- (2) A implementação atual do compilador didático aceita apenas as operações de soma e multiplicação. Modifique o compilador para que aceite também subtração e divisão. Para essas operações, gerar as instruções SUB e DIV.
- (3) O compilador didático possui um back-end que gera código para uma máquina de pilha hipotética:

```
CodeGen backend = new CodeGen();  
String codigo = backend.geraCodigo(arv);
```

O *back-end* recebe como entrada a árvore sintática e percorre essa árvore gerando código para uma máquina de pilha. Substituir o back-end desse compilador por um back-end *interpretador*, ou seja, ao invés de gerar uma String com código para ser executado na máquina de pilha, o back-end já devolve o resultado da computação da expressão de entrada.

- (4) Implementar em Java, uma máquina de pilha que execute as instruções geradas pelo compilador didático. A máquina de pilha deve receber como entrada um arquivo texto com as instruções e devolver como saída o resultado final da execução dessas instruções. A classe principal deve se chamar `MaquinaPilha`. Ex:

```
~$ java MaquinaPilha arquivoDeEntrada
```

OBS: Os exercícios 1, 2 e 3 são incrementais, ou seja, o exercício 2 deve ser implementado em cima das modificações feitas no exercício 1, e o exercício 3 deve ser implementado em cima da versão gerada no exercício 2.