

## Trabalho Prático II – Análise Sintática

### Descrição do trabalho

Nesta etapa, você deverá implementar um analisador sintático para a linguagem *Javazim*, cuja descrição encontra-se em anexo.

Seu analisador sintático deverá ser implementado em linguagem *Java* com o auxílio da ferramenta *Java CUP*, disponível para download no link <http://www2.cs.tum.edu/projects/cup/>, além da ferramenta *JFlex*, disponível em <http://jflex.de>.

O analisador sintático deverá reportar possíveis erros ocorridos no programa-fonte, informando qual o erro encontrado e sua localização no arquivo-fonte.

Para implementar o analisador sintático, você deverá modificar a estrutura gramatical da linguagem fornecida na descrição do primeiro trabalho prático. Você deverá adequá-la às restrições de precedência e associação de operadores, eliminando os conflitos. As regras de precedência e associação de operadores pode ser as mesmas da linguagem *Java* (pesquise a tabela de precedência em fóruns confiáveis ou nos documentos técnicos de *JavaDoc* na documentação da Sun/Oracle).

Alguns ajustes em seu analisador léxico poderão ser necessários a fim de possibilitar a correta integração do mesmo com o analisador sintático.

### O que entregar?

Você deverá entregar nesta etapa:

1. A nova versão da gramática (em papel);
2. O arquivo `.cup` (em papel e por e-mail [adriano.santos@ifmg.edu.br](mailto:adriano.santos@ifmg.edu.br));
3. Programa com todos os arquivos-fonte (será apresentado em laboratório) (por e-mail [adriano.santos@ifmg.edu.br](mailto:adriano.santos@ifmg.edu.br));
4. Testes realizados com programas corretos e errados (no mínimo, 3 certos e 3 errados) (em papel e por e-mail [adriano.santos@ifmg.edu.br](mailto:adriano.santos@ifmg.edu.br)).

Para avaliar a correção, o programa deverá mostrar as regras que foram utilizadas.