

Criação de Tutoriais Iterativos e Testes para Compiladores

18. Compiladores

Prof. Rogério Aparecido Gonçalves

Campo Mourão

2020

SUMÁRIO

RESUMO GERAL	3
ESCOPO E OBJETIVO DA PROPOSTA	3
RESULTADOS E MATERIAIS DIGITAIS PREVISTOS	3
METODOLOGIA PROPOSTA	5
FASE 01 – DURANTE O DESENVOLVIMENTO (MARÇO A JUNHO)	5
FASE 02 – APÓS O DESENVOLVIMENTO	5
PAPÉIS E RESPONSABILIDADES	5

RESUMO GERAL

A disciplina de Compiladores é uma disciplina básica para a Computação. Nela os alunos tem contato por meio do desenvolvimento de trabalhos com o projeto e desenvolvimento de uma ferramenta de compilação, um compilador. O projeto é dividido, normalmente, em quatro fases que correspondem às fases do processo de compilação, sendo elas Análise Léxica, Análise Sintática, Análise Semântica e Geração de Código. A disciplina de compiladores, pode ser considerada uma disciplina difícil em termos de conteúdo e das implementações dos projetos. O desenvolvimento da ferramenta de compilação utilizando a linguagem de programação Python, com bibliotecas como o PLY e llvmlite, facilita a implementação dos Analisadores Léxicos (lex) e Sintáticos (yacc) e a geração de código intermediário para LLVM, e possibilita que a infraestrutura e ferramentas do LLVM possam ser utilizadas na geração do código final e executável. Com a popularização de notebooks iterativos em Jupyter, que permitem o aluno testar trechos de código, modificar e acrescentar novas funcionalidades e testar novamente. Na disciplina tem sido desenvolvidos tutoriais para cada uma das fases de compilação, no intuito de facilitar o desenvolvimento dos projetos, fornecendo um código inicial de referência para o desenvolvimento de cada fase.

ESCOPO E OBJETIVO DA PROPOSTA

O objetivo desse projeto é o desenvolvimento de tutoriais no formato de notebooks Jupyter e de conjuntos de testes que possam ser executados de maneira automática em plataformas de competições de programação.

Os tutoriais nesse formato ajudam os alunos a não precisarem instalar de imediato todas as ferramentas necessárias para o desenvolvimento do projeto, possibilitando que os alunos testem os exemplos e trechos de códigos e verifiquem o funcionamento de maneira iterativa via *browser*.

A disciplina que será atendida é a de Compiladores dos cursos de Bacharelado em Ciência da Computação e Engenharia da Computação.

RESULTADOS E MATERIAIS DIGITAIS PREVISTOS

Os artefatos que serão criados são o repositório do projeto no Github¹ com os tutoriais e os mesmos notebooks Jupyter que poderão ser executados localmente pelo aluno, mas estarão disponíveis na plataforma Google Colaboratory² para execução remota, sem a necessidade de instalar as ferramentas e bibliotecas em seu computador. O Código 1 apresenta o exemplo de cálculo do Fatorial de um número na linguagem TPP.

Código 1: Exemplo de Código em TPP

```
inteiro: n
flutuante: a[10]

inteiro fatorial(inteiro: n)
  inteiro: fat
  se n > 0 então {não calcula se n > 0}
  fat := 1
  repita
    fat := fat * n
    n := n - 1
  até n = 0
  retorna(fat) {retorna o valor do fatorial de n}
```

¹ <https://github.com/rogerioag/tutorial-de-compiladores>

² <https://colab.research.google.com>

```

senão
  retorna(0)
fim

inteiro principal()
  leia(n)
  escreva(fatorial(n))
  retorna(0)
fim

```

As Figuras 01 e 02 apresentam o *notebook* desenvolvido para a Análise Léxica.

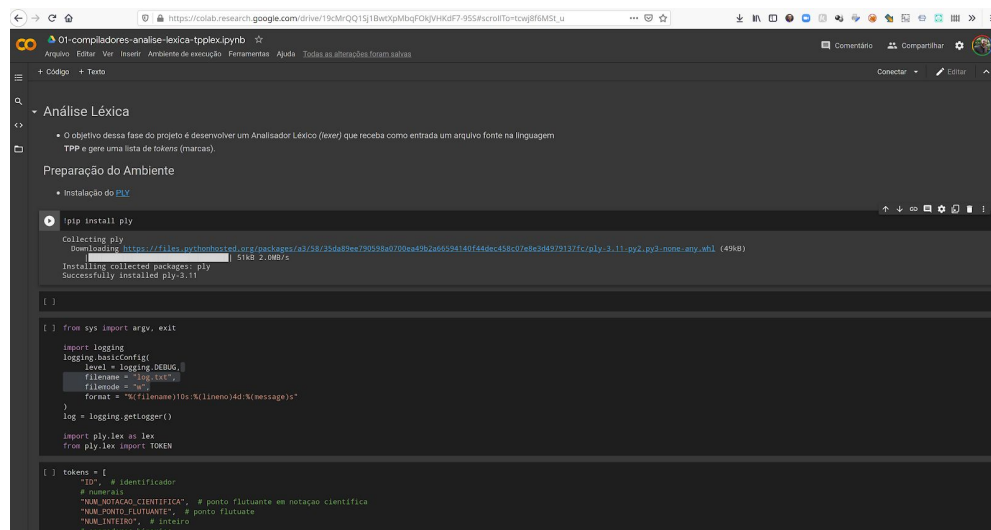


Figura 01: Notebook Jupyter para o Analisador Léxico

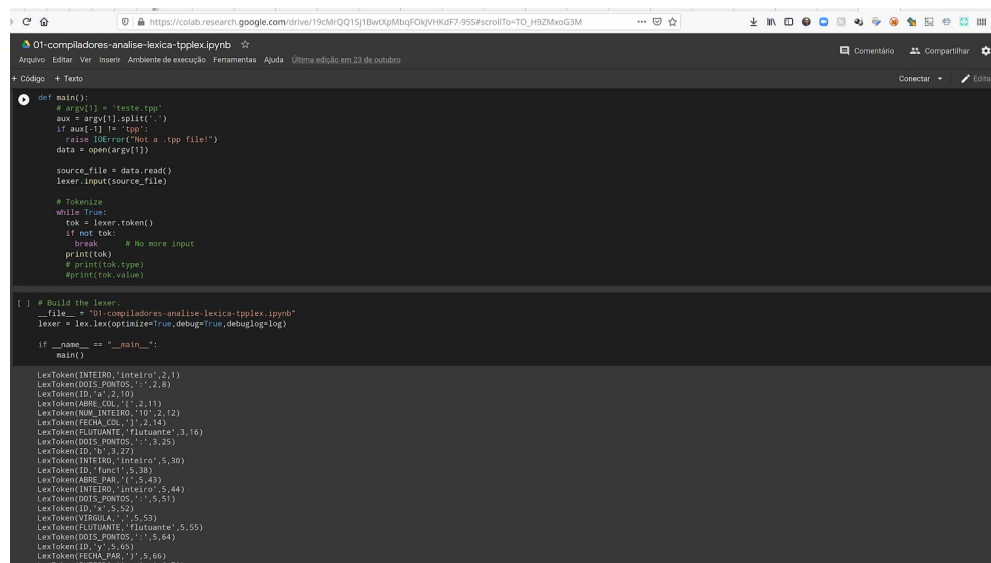


Figura 02: Notebook Jupyter para o Analisador Léxico

Os testes automatizados serão disponibilizados no repositório do projeto no Github e pretendemos gerar para o BOCA³ inicialmente. O BOCA é um sistema utilizado em maratonas de programação para a análise e julgamento das soluções de problemas. A ideia é produzir testes para os analisadores Léxico, Sintático e Semântico e para a Geração de Código. Os alunos durante o desenvolvimento poderão executar uma bateria de testes para a cobertura de diversos aspectos da linguagem objeto de estudo na disciplina, a linguagem TPP. Atualmente um conjunto de testes para

³ <https://github.com/cassiopc/boca/>

cada fase do desenvolvimento é disponibilizado no moodle e os alunos executam manualmente. O relatório da execução na plataforma facilitará o processo de avaliação dos trabalhos da disciplina.

METODOLOGIA PROPOSTA

A disponibilização de tutoriais iterativos que o aluno possa executar e alterar o código de exemplos e de código de start do projeto, é interessante pois permite o aluno perceber os impactos e efeitos das alterações que faz nos trechos de código. Os notebooks Jupyter possibilitam que tais execuções sejam feitas de imediato e a visualização de resultados apareçam diretamente em um *browser*, sem a necessidade de instalação local de um conjunto de ferramentas.

SELEÇÃO DO BOLSISTA

Será necessário um bolsista que desenvolva em Python e que já tenha concluído a disciplina de Compiladores do Curso de Bacharelado em Ciência da Computação.

FASE 01 – DURANTE O DESENVOLVIMENTO (MARÇO A JUNHO)

Descrever o planejamento de trabalho para 4 meses:

Etapa 1 - Término do notebook para Análise Sintática (Marco)

Etapa 2 - Término do notebook para Análise Semântica (Marco)

Etapa 3 - Desenvolvimento do notebook para Geração de Código (Abril)

Etapa 4 - Criação de testes na plataforma BOCA para Análises Léxica, Sintática e Semântica e para a Geração de Código (Maio/Junho)

Etapa 5 - Escrita do Relatório Final do projeto (Junho)

FASE 02 – APÓS O DESENVOLVIMENTO

Os tutoriais serão disponibilizados no repositório do projeto no Github e nos links dos notebooks Jupyter de plataformas como o Google Colaboratory.

Os testes serão elaborados para a plataforma BOCA, inicialmente. Pretendemos disponibilizar os fontes para a criação dos testes.

PAPÉIS E RESPONSABILIDADES

Equipe Executora: tutores, servidores colaboradores, discentes bolsistas, parceiros externos.

[illegible]