

Lógica da Computação

Aula 01

Raul Ikeda

1º semestre de 2020

Objetivos de Aprendizagem

1. Especificar uma gramática para reconhecer uma linguagem de interesse.
2. Saber programar e gerar o diagrama explicativo de um autômato que implementa uma gramática.
3. Compreender os conceitos básicos sobre Lógica Matemática, Teoria da Computabilidade e Máquina de Turing.
4. Entender todos os passos necessários à compilação de programas.

Mapeamento das Avaliações

- LING - LINGUAGEM:
 - Especificar uma gramática para reconhecer uma linguagem de interesse.
- COMP - COMPILADOR:
 - Saber programar e gerar o diagrama explicativo de um autômato que implementa uma gramática.
 - Entender todos os passos necessários à compilação de programas.
- PROV - PROVAS:
 - Especificar uma gramática para reconhecer uma linguagem de interesse.
 - Saber programar e gerar o diagrama explicativo de um autômato que implementa uma gramática.
 - Compreender os conceitos básicos sobre Lógica Matemática, Teoria da Computabilidade e Máquina de Turing.

Avaliações: LING - Linguagem

- Durante o curso será desenvolvida uma linguagem de programação com características proprietárias.
- Projeto em caráter de APS e individual.
- Requisitos:
 - i. Adequar a linguagem à uma GLC e estruturá-la segundo o padrão EBNF.
 - ii. Utilizar as ferramentas Flex e Bison (ou semelhantes) para realizar as etapas de Análise Léxica e Sintática.
 - iii. Utilizar a LLVM (ou semelhantes - incluindo o próprio compilador) para implementar a sua linguagem até a fase final de compilação. Não é preciso implementar um compilador novo.
 - iv. Criar um exemplo de testes que demonstre as características da sua Linguagem.
 - v. Fazer uma apresentação de 15 minutos na data da entrega final.

Avaliações: LING - Linguagem

- Nota da avaliação:
 - A+: se cumprir todos os requisitos acima.
 - B: se o requisito número 3 for feito com o próprio compilador.
 - D: se não cumprir os requisitos.
 - I: se não houver entrega ou se for irrelevante.

Haverá 2 entregas parciais mais uma final. Cada atraso implica em perda de $\frac{1}{2}$ conceito na avaliação.

Avaliações: COMP - Compilador

- Durante o semestre será desenvolvido incrementalmente um Compilador da linguagem **Matlab**, aplicando o estudo de compiladores. No total serão 10 atividades com entregas.
- Requisitos:
 - Individual. Uso do Git **privado** obrigatório.
 - Desenvolvimento em aula estúdio. As tarefas pendentes terão prazos de entrega entre 1 e 2 semanas dependendo do roteiro.
 - Usar uma linguagem orientada a objetos com recursividade e dicionários.

Avaliações: COMP - Compilador

- Nota:
 - I se não entregar nenhuma atividade.
 - D se não entregar pelo menos uma atividade.
 - A se não atrasar nenhuma entrega.
 - Para cada atraso na entrega, desconta-se $\frac{1}{2}$ conceito.
 - 11^a Entrega extra: adiciona $\frac{1}{2}$ conceito.
 - Será permitido o atraso de até 1 relatório sem ônus na nota.

Avaliações: PROV - Prova

- 40% - AI:
 - Teoria de Conjuntos.
 - Linguagens Regulares.
 - Linguagens Livres de Contexto.
 - Autômatos Finitos e de Pilha.
 - Hierarquia de Chomsky.
 - Linguagens Sensíveis ao Contexto.
 - Máquina de Turing.
 - Linguagens Recursivamente Enumeráveis.
- 60% - AF:
 - Computabilidade e Decidibilidade.
 - Complexidade e Intratabilidade.
 - Prova Matemática.
 - Lógica Proposicional e de Predicados.
 - Verificação de programas.

Nota Final

- Se $LING$, $COMP$ e $PROV \geq 5$ ou C :

$$NF = 0,2LING + 0,4COMP + 0,4PROV$$

- Caso contrário:

$$NF = \min(LING, COMP, PROV)$$

Será usada a tabela oficial de conversão de conceito para nota do Blackboard.

Bibliografia

- Compiladores:
 - AHO, A. V.; LAM, M. S.; SETHI, R.; ULLMAN, J., Compiladores: Princípios, Técnicas e Ferramentas, 2ª ed., Longman, 2007
 - JOSÉ NETO, J., Introdução à Compilação., 1ª ed., Elsevier, 2016
- Linguagens Formais e Autômatos:
 - RAMOS, M. V. M.; JOSÉ NETO, J.; VEJA, I. S., Linguagens Formais. Teoria, Modelagem e Implementação, 1ª ed., Bookman, 2009
 - HOPCROFT, J. E.; ULLMAN, J. D.; MOTWANU, R., Introdução à Teoria dos Autômatos, Linguagens e Computação, 1ª ed., CAMPUS, 2002
- Computabilidade e Lógica:
 - BOOLOS, G. S.; BURGESS, J. P.; JEFFREY, R. C., Computabilidade e Lógica, 1ª ed., Unesp, 2012
 - SILVA, F. C.; FINGER, M.; MELO, A. C. V., Lógica para Computação, 2ª ed., Cengage, 2017
- Complexidade:
 - CORMEN, T. H.; LEISERSON, C. E.; RIVEST, R. L.; STEIN, C., Algoritmos: teórica e prática., 3ª ed., Elsevier-Campus., 2012
 - SIPSER, M., Introdução à Teoria da Computação, 2ª ed., Thomson Pioneira, 2007

Próxima Aula

- Discussão do artigo: José Neto, J., A Teoria da Computação e o profissional de informática. Revista de Computação e Tecnologia da PUC-SP, vol. 1, 2009.
 - Revista Open Access, mas publicado no Blackboard.
 - Duração: até 30 minutos no início.
- Teoria dos Conjuntos.
- Gramáticas e Linguagens

Referências:

- Ramos et al. Cap 1 e 2
- Hopcroft et al. Cap 1.5
- Sipser Cap 0.3

Atividade

- Introdução a Compiladores.
- Roteiro Zero do Compilador.
- Lembrando:
 - Individual.
 - Usar uma linguagem orientada a objetos com recursividade e dicionários.