



UNIVERSIDADE FEDERAL DE OURO PRETO  
DEPARTAMENTO DE COMPUTAÇÃO



PLANO DE ENSINO

|  |   |   |
|--|---|---|
| <b>Nome do Componente Curricular em português:</b><br>Construção de Compiladores I   |   | <b>Código:</b><br>BCC328                              |
| <b>Nome do Componente Curricular em inglês:</b><br>Compiler Construction I   |   |   |
| <b>Nome e sigla do departamento:</b><br>Departamento de Computação (DECOM)   |   | <b>Unidade acadêmica:</b><br>ICEB                     |
| <b>Nome do docente:</b><br>José Romildo Malaquias  |   |   |
| <b>Carga horária semestral:</b><br>60 horas  | <b>Carga horária semanal teórica:</b><br>4 horas/aula | <b>Carga horária semanal prática:</b><br>0 horas/aula |
| <b>Data de aprovação na assembleia departamental:</b><br>20/08/2021  |   |   |
| <b>Ementa:</b><br>Implementação de linguagens de programação: compilação e interpretação; análise léxica; análise sintática; análise semântica.  |   |   |
| <b>Conteúdo Programático:</b> <ul style="list-style-type: none"><li>• Implementação de linguagens de programação: compilação e interpretação</li><li>• Análise léxica: análise léxica, analisador léxico ad hoc, expressões regulares, autômatos finitos e geradores de analisadores léxicos</li><li>• Análise sintática: análise sintática, gramáticas livre de contexto, análise sintática descendente, análise sintática ascendente, geradores de analisadores sintáticos e ações semânticas</li><li>• Análise semântica: árvore abstrata, tabelas de símbolos, checagem de tipos e regras semânticas</li></ul> |   |   |
| <b>Objetivos:</b> <p>Ao final do curso é esperado que o aluno:</p> <ul style="list-style-type: none"><li>- possua uma visão geral sobre o processo de compilação sob o ponto de vista de implementação</li><li>- saiba especificar aspectos léxicos, sintáticos e semânticos de linguagens através de expressões regulares e gramáticas livres de contexto</li><li>- conheça as principais técnicas e ferramentas de apoio usadas na construção de compiladores, sabendo usá-las na especificação e implementação de linguagens de programação</li></ul>   |   |   |
| <b>Metodologia:</b> <p><b>Aulas expositivas</b> usando vídeos online <i>preferencialmente assíncronas</i> para apresentação do conteúdo, e disponibilizados para os alunos.</p> <p><b>Exercícios de fixação</b> para explorar a base teórica do conteúdo programático.</p> <p><b>Exercícios práticos</b> para aplicação de técnicas de compilação na construção do <i>front end</i> de um compilador para uma linguagem de programação simples.</p>  |   |   |

**Provas** escritas.

**Recursos necessários** (eventualmente não limitado a estes):

- Computador ou notebook: utilizado para acesso aos vídeos, documentos eletrônicos e demais softwares necessários para a realização das tarefas de estudo e programação. Embora o acesso à maioria dos recursos seja viável através de aparelhos de celular e existam aplicativos para se programar neste tipo de aparelho, as tarefas de programação em dispositivos móveis são de execução mais trabalhosa devido às limitações nos recursos de digitação e visualização de resultados inerentes a essa plataforma.
- Acesso à internet: o acesso à internet é essencial para fazer download e upload de arquivos para estudo e realização das tarefas da disciplina e participação de atividades síncronas de esclarecimento de dúvidas e/ou avaliações online.
- Navegador de internet: para acesso aos conteúdos online, sistema acadêmico Moodle, e várias ferramentas online (armazenamento na nuvem, documentos, planilhas, vídeo/áudio conferências) e outros conteúdos.
- Ferramentas de desenvolvimento: ambiente específico para a tarefa de programação de um compilador na linguagem OCaml no sistema operacional Linux, disponíveis livremente para download e instalação.

**Atividades avaliativas:**

**Primeira avaliação** (30% nota total)

- Temas: introdução, análise léxica, análise sintática descendente
- Prova escrita 1
- No dia 17/11/2021, no horário da aula, pela sala virtual no Moodle

**Segunda avaliação** (30% da nota total)

- Temas: análise sintática ascendente, análise semântica
- Prova escrita 2
- No dia 05/01/2022, no horário da aula, pela sala virtual no Moodle

**Terceira avaliação** (40% da nota total)

- Tema: implementação do *front end* de um compilador
- Diversos exercícios práticos
- A nota será a média dos exercícios
- Submissão pela plataforma GitHub Classroom

**Exame especial**

- Prova escrita e/ou trabalho incluindo conteúdo teórico e/ou prático
- No dia 12/01/2022, no horário da aula, pela sala virtual no Moodle e/ou plataforma Github Classroom
- Exame parcial substitui uma das três avaliações
- Exame total substitui todas as três avaliações

**Cronograma:**

| <b>Aula</b> | <b>Data</b> | <b>Dia</b> | <b>Conteúdo</b>   |
|-------------|-------------|------------|---|
| 1           | 20 set      | seg        | Apresentação da disciplina; Compilação e interpretação (síncrona) |

|    |        |     |   |
|----|--------|-----|---|
| 2  | 22 set | qua | Linguagem de implementação; Ferramentas                                     |
| 3  | 27 set | seg | Análise léxica: introdução  |
| 4  | 29 set | qua | Análise léxica: geradores   |
| 5  | 04 out | seg | Análise léxica: projeto   |
| 6  | 06 out | qua | Análise léxica: expressões regulares  |
|    | 11 out | seg | (Recesso acadêmico)   |
| 7  | 13 out | qua | Análise léxica: autômatos finitos   |
| 8  | 18 out | seg | Análise léxica: analisador baseado em autômatos finitos                     |
| 9  | 20 out | qua | Análise sintática: introdução   |
| 10 | 25 out | seg | Análise sintática: gerador  |
| 11 | 27 out | qua | Análise sintática: projeto  |
|    | 01 nov | seg | (Recesso acadêmico)   |
| 12 | 03 nov | qua | Análise sintática: análise descendente recursiva                            |
| 13 | 08 nov | seg | Análise sintática: <i>nullable</i> , conjuntos <i>first</i> e <i>follow</i> |
| 14 | 10 nov | qua | Análise sintática: tabelas LL(0) e LL(1)                                    |
|    | 15 nov | seg | (Feriado Nacional: Proclamação da República)                                |
| 15 | 17 nov | qua | <b>Prova 1</b>  |
| 16 | 22 nov | seg | Análise sintática: análise ascendente e autômato de pilha                   |
| 17 | 24 nov | qua | Análise sintática: tabelas LR(0) e SLR                                      |
| 18 | 29 nov | seg | Análise sintática: tabelas LR(1) e LALR(1)                                  |
| 19 | 01 dez | qua | Análise sintática: resolução de conflitos                                   |
| 20 | 06 dez | seg | Árvores de sintaxe abstrata   |
| 21 | 08 dez | qua | Análise semântica: introdução, projeto                                      |
| 22 | 13 dez | seg | Análise semântica: tabela de símbolos                                       |
| 23 | 15 dez | qua | Análise semântica: regras de análise semântica                              |
|    | 20 dez | seg | (Recesso acadêmico)   |
|    | 22 dez | qua | (Recesso acadêmico)   |
|    | 27 dez | seg | (Recesso acadêmico)   |
|    | 29 dez | qua | (Recesso acadêmico)   |
| 24 | 03 jan | seg | Análise semântica: regras de análise semântica                              |
| 25 | 05 jan | qua | <b>Prova 2</b>  |
| 26 | 12 jan | qua | <b>Exame especial</b>   |

#### Bibliografia Básica:

- A. V. Aho, M. S. Lam, R. Sethi, J. D. Ulman. **Compiladores: Princípios, Técnicas e Ferramentas**. 2. edição. Pearson, 2007. [Disponível online em MinhaUFOP/BVirtual Pearson](#).
- Santos, Pedro Reis; Langlois, ThiBault. **Compiladores: da Teoria à Prática**. LTC, 2018. [Disponível online em MinhaUFOP/Minha Biblioteca](#).
- Louden C. **Compiladores: Princípios e Prática**. Cengage, 2004. [Disponível online em MinhaUFOP/Minha Biblioteca](#).

#### Bibliografia Complementar:

- Appel, Andrew W. **Modern Compiler Implementation in ML**. Cambridge University Press, 1998.
- Mogensen, Torben Ægidius. **Introduction to Compiler Design**. 2. edição. Springer, 2011.

- Grüne, Dick; Reeuwijk, Kees van; Bal, Henri E.; Jacobs, Cerial J. H.; Langendoen, Koen. **Modern Compiler Design**. 2. edição. Springer, 2012.
- Wilhelm, Reinhard; Seidl, Helmut; Hack, Sebastian. **Compiler Design: Syntactic and Semantic Analysis**. Springer, 2013.
- Fischer, Charles N.; Cytron, Ron K.; LeBlanc, Richard J. Jr. **Crafting a Compiler**. Pearson, 2010.
- Cooper, Keith D; Torczon, Linda. **Engineering a Compiler**. 2. edição. Elsevier, 2011. ISBN-13: