



UNIVERSIDADE
FEDERAL DO CEARÁ

UNIVERSIDADE FEDERAL DO CEARÁ
PRÓ-REITORIA DE GRADUAÇÃO
COORDENADORIA DE PROJETOS E ACOMPANHAMENTO CURRICULAR
DIVISÃO DE DESENVOLVIMENTO CURRICULAR

FORMULÁRIO PARA CRIAÇÃO E/OU REGULAMENTAÇÃO DE DISCIPLINA

() **Regulamentação** (se a disciplina está prevista no Projeto Pedagógico)

() **Criação/Regulamentação** (se a disciplina não está prevista no Projeto Pedagógico)

1. Unidade Acadêmica que oferta a Disciplina (Faculdade, Centro, Instituto, *Campus*):
Campus Quixadá

2. Departamento que oferta a Disciplina (quando for o caso):

--

3. Curso(s) de Graduação que oferta(m) a disciplina

Código do Curso	Nome do Curso	Grau do Curso ¹	Currículo (Ano/Semestre)	Caráter da Disciplina ²	Semestr e de Oferta ³	Habilitação ⁴
401	Sistemas de Informação	Bacharelado	2007.2A	Optativa	8º	--
402	Engenharia de Software	Bacharelado	2010.1	Optativa	6º	--
404	Ciência da Computação	Bacharelado	2013.1	Obrigatória	5º	--

4. Nome da Disciplina:

Compiladores

5. Código da Disciplina (preenchido pela PROGRAD):

QXD0025

6. Pré-Requisitos	Não ()	Sim (X)	
		Código	Nome da Disciplina/Atividade
		QXD0040 (404)	Linguagens Formais e Autômatos
		QXD0016 (401,402)	Linguagens de Programação

7. Correquisitos

Sim ()

¹ Preencher com *Bacharelado, Licenciatura* ou *Tecnólogo*.

² Preencher com *Obrigatória, Optativa* ou *Eletiva*.

³ Preencher quando obrigatória.

⁴ Quando eletiva, preencher com a habilitação ou ênfase a que se vincula a disciplina.

	Não (X)	Código	Nome da Disciplina/Atividade

8. Equivalências	Não (X)	Sim ()	
		Código	Nome da Disciplina/Atividade

9. Turno da Disciplina (é possível marcar mais de um item):
 Matutino Vespertino Noturno

10. Regime da Disciplina:
 Semestral Anual Modular

11. Justificativa para a criação/regulamentação desta disciplina – Máximo de 500 caracteres
 (mostrar a importância da área / do conteúdo para a formação do aluno, a pertinência da disciplina na integralização curricular e outros aspectos):
 Os compiladores transformam código escrito em uma linguagem de computador (linguagem fonte) em outra linguagem de computador (linguagem alvo), a qual assume, frequentemente, a forma binária (código objeto). Dessa forma, permitem a implantação de sistemas que, sem um compilador, seriam inviáveis ou muito caros e demorados para implementar. São diversas as tarefas de um compilador e esta disciplina pretende prover ao aluno o conhecimento necessário para entender tais tarefas e desenvolver no aluno a habilidade de implementá-las. O conhecimento adquirido na disciplina permitirá ao aluno criar uma linguagem e implementar um compilador para ela.

12. Objetivo(s) da Disciplina:

Objetivo geral: Dar subsídio para que o aluno compreenda os conceitos básicos de compiladores, seus componentes e funcionamento para ajudá-lo a ter uma visão mais aprofundada no desenvolvimento de sistemas com qualidade na utilização dos recursos computacionais e também possibilitar a implementação de um compilador.

Objetivos específicos:

- Fornecer ao aluno informações sobre os conceitos e a organização interna dos compiladores;
- Apresentar a construção de linguagens de programação através da especificação dos seus aspectos léxicos e sintáticos;
- Apresentar as tarefas de um compilador e seu funcionamento, em especial análise sintática, análise semântica e geração de código;
- Mostrar técnicas e ferramentas de apoio usadas na construção de compiladores;
- Mostrar os problemas que podem acontecer em um programa em função das características de um compilador.

13. Ementa:
 Introdução a Compiladores, Análise Léxica, Análise Sintática e Abstração de Sintaxe, Análise Semântica, Tabela de Símbolos e Análise de Escopo; Seleção de Instruções; Análise de Longevidade; Seleção de Registradores; Emissão de Código, Tópicos Especiais em Compiladores.

14. Descrição da Carga Horária				
Número de Semanas: 16	Número de Créditos: 4	Carga Horária Total: 64	Carga Horária Teórica: 32	Carga Horária Prática: 32

15. Bibliografia Básica (sugere-se a inclusão de, pelo menos, 03 títulos):

AHO, Alfred V.; LAM, Monica S.; SETHI, Ravi; ULLMAN, Jeffrey D., Compiladores: princípios, técnicas e ferramentas, Pearson, 2a Edição, 2007.

LOUDEN, Kenneth. Compiladores: princípios e práticas. Thomson Pioneira, 2004.

HOPCROFT, John E. Introdução à teoria de autômatos, linguagens e computação. Rio de Janeiro: Elsevier, 2002. 560 p

...

MAK, Ronald, Writing Compilers and Interpreters: A Software Engineering Approach, Wiley, 3rd Edition, 2009.

16. Bibliografia Complementar (sugere-se a inclusão de, pelo menos, 05 títulos – de acordo com instrumento de avaliação de Curso de Graduação, INEP/maio-2012 ou legislação posterior):

DELAMARO, M. E. Como Construir Um Compilador Utilizando Ferramentas Java, 1 ed, 2004
Novatec

MENEZES, Paulo Blauth. Linguagens formais e autômatos. 6. ed. Porto Alegre: Bookman, 2011. 215 p. (Livros didáticos ; n.3 Série Livros Didáticos ; 3) ISBN: 9788577807659

SEBESTA, Robert W. Conceitos de linguagens de programação. 5. ed. Porto Alegre: Bookman, 2003. ISBN 8536301716.

FORBELLONE, André Luiz Villar; EBERSPÄCHER, Henri Frederico. Lógica de programação: a construção de algoritmos e estruturas de dados . 3. ed. São Paulo: Makron, 2005. xii, 218 p.

...

GRUNE, D., Jacobs, C., Parsing Techniques: A Practical Guide (Monographs in Computer Science), Springer, 2nd Edition, 2008. ISBN: 9780387202488

LEVINE, J., Flex & Bison: Text Processing Tools. O'Reilly Media, 2009. ISBN: 9780596155971

MOZGOVOY, Maxim, Algorithms, Languages, Automata, & Compilers: A Practical Approach, Jones and Bartlett Publishers, Inc., 2009. ISBN: 9780763776275

TOSCANI, S. S., Implementação de Linguagens de Programação – Compiladores, 1 ed, 2008
Bookman. ISBN: 9788577803484

17. Aprovação do Colegiado do Departamento (quando for o caso)

Data de Aprovação:

Chefe(a) do Departamento
Assinatura e Carimbo

18. Aprovação do(s) Colegiado(s) de Curso(s)

Código do Curso:

Data de Aprovação:

Coordenador(a) do Curso
Assinatura e Carimbo

Código do Curso:	Data de Aprovação:	<hr/> Coordenador(a) do Curso Assinatura e Carimbo
Código do Curso:	Data de Aprovação:	<hr/> Coordenador(a) do Curso Assinatura e Carimbo

19. Aprovação do Conselho da Unidade Acadêmica	
Data de Aprovação:	<hr/> Diretor(a) da Unidade Acadêmica Assinatura e Carimbo

20. Aprovação do Conselho de Ensino, Pesquisa e Extensão (Câmara de Graduação)	
Data de Aprovação:	<hr/> Presidente(a) da Câmara de Graduação Assinatura e Carimbo

Orientações para tramitação do processo:

Deve ser aberto e encaminhado processo à Pró-Reitoria de Graduação / Câmara de Graduação, contendo: 1) Ofício(s) informando a data de aprovação da criação e/ou regulamentação da(s) disciplina(s) pela Coordenação do Curso, pelo(s) Departamento(s) envolvido(s) – se for o caso – e pela Direção da Unidade Acadêmica; 2) Formulário para Criação e/ou Regulamentação de Disciplina integralmente preenchido, com assinaturas, datas e carimbos solicitados.

ANEXO - Descrição do Conteúdo e Carga Horária

Descrição do Conteúdo e Carga Horária			
Unidades e Assuntos das Aulas	Nº de Horas Teóricas	Nº de Horas Práticas	Nº de Horas EaD (quando for o caso):

1.Introdução aos Compiladores Interpretador vs Compilador, Estrutura do Compilador, Processo de Compilação, Montador e Cross-Compiler			6	-	-
2.Análise Léxica Tokens, autômatos NFA e DFA e Expressões Regulares			6	10	-
3.Análise Sintática Analisador LL, Analisador LR, Analisador LALR, Árvore Sintática			6	10	-
4.Análise Semântica Árvore Semântica			6	6	-
5.Código Intermediário e Otimização			4	-	-
6.Geração de Código			4	6	-
Número de Semanas:	Número de Créditos:	Carga Horária Total:	Carga Horária Teórica:	Carga Horária Prática:	Carga Horária EaD:
16	4	64	32	32	0