

<b>Dados de identificação</b>		
<i>Disciplina</i>	<b>Especificação Formal N</b>	
<i>Oferecida para</i>	Bacharelado em Ciência da Computação e Engenharia de Computação	
<i>Período Letivo</i>	2020/2	
<i>Professor Responsável</i>	Daltró José Nunes	
<i>Sigla</i>	INF01001	
<i>Carga horária (horas)</i>	60h	
<i>CH Autônoma (horas)</i>	10	
<i>CH Coletiva (horas)</i>	50	
<i>CH Individual (horas)</i>	0	
<b>Súmula</b>	Revisão matemática; métodos e linguagens; classificação dos métodos formais; especificação executáveis (prototipação); apresentação e aplicação de métodos e linguagens de especificação formal: VDM, Z, CCS, CSP, OBJ, LOTOS, UNIT e outros.	
<b>Objetivos</b>	<p>O objetivo dos cursos de Engenharia em geral é dotar os profissionais de habilidades e competências para encontrar solução para problemas, que normalmente envolvem elementos da natureza. Os princípios, métodos ou processos da engenharia, mostram as fases necessárias para resolver problemas.</p> <p>Métodos e processos podem também ser usados para resolver problemas abstratos. De forma bem resumida, a engenharia de software começa pela (i) análise de requisitos, depois vem as fases de (ii) especificação formal da solução, (iii) proposta de projeto e finalmente, (iv) proposta de implementação.</p> <p>Muitos profissionais de engenharia de software começam pela análise de requisitos e seguem diretamente para a fase de implementação. O problema deste processo está na interpretação ambígua dos requisitos. Testes são necessários para verificar se a implementação está correta. Entretanto, em se tratando de sistemas críticos, tais como controle de centrais nucleares, sistemas de monitoramento e exames de pacientes, balísticas, aviônica etc. testes, mesmo com uma boa cobertura, não são suficientes.</p> <p>Conforme processo recomendado pela Engenharia de Software, usando um método formal, a solução do problema é formalizada com a construção de uma especificação formal (modelo matemático), que independe de implementação. Verificação de modelo (Model Checking) consiste na definição e prova de propriedades da especificação, validando a especificação. Depois, usando criatividade, um programa é construído a partir da especificação, usando uma linguagem de programação, por exemplo, Java. Verificação de equivalência (Equivalence checking) consiste na prova de que o programa satisfaz a especificação. Essas provas são complexas para serem realizadas a mão e provadores automáticos de teoremas são aplicados, como, por exemplo, SPEC#. Provas formais garante que o programa satisfaz sua especificação dispensando o uso de testes.</p> <p>Dependendo do problema, existem métodos formais para especificação de problemas sequenciais, concorrentes (inter leaving) e paralelos de tempo real.</p> <p>Muitos dos métodos formais possuem um interpretador e, assim, especificações formais podem ser executadas contribuindo para sua validação junto ao cliente (antes da construção do programa). Especificações executáveis são chamadas de protótipo do sistema. Existe um lema da Engenharia de Software estabelecendo que quanto mais tarde for encontrado um erro no processo de software, mais caro custa a correção do mesmo.</p> <p>Relação com outras disciplinas: Especificação formal é importante para a disciplina de engenharia de software ou para qualquer disciplina que pretende implementar conceitos como: Parses (Compiladores), Gerenciamento de Memória e Escalonadores (Sistemas Operacionais), Protocolos (Redes), Modelos de Banco de Dados etc.</p>	
<b>Conteúdo Programático</b>		
<b>Título</b>	<b>Conteúdo</b>	<b>Semana</b>
Introdução	Revisão Matemática e classificação dos métodos formais.	1
MAUDE	Aulas expositivas para apresentação do método (linguagem) MAUDE	2 a 10
Método CCS	Aulas expositivas para apresentação do CCS.	11 a 12
Método VDM	Aulas expositivas para apresentação do método VDM	13 a 14
Apresentação de Seminários	Aula para apresentação dos Seminários pelos alunos da pós-graduação com a frequência obrigatória dos alunos da	15
<b>Metodologia</b>		
<i>Estratégias didáticas em atividades remotas</i>	As atividades ocorrerão de forma síncrona ou assíncrona. Quando as atividades forem síncronas, elas ocorrerão nos horários regulares da disciplina. Esses encontros serão gravados e disponibilizados para uso posterior dos alunos. Nessas aulas, instruções serão dadas de forma síncrona ou assíncronas, materiais previamente gravados, ou por escrito. Os alunos trabalharão em seus computadores pessoais, e o professor estará disponível de forma síncrona para esclarecimento de dúvidas. As atividades assíncronas consistem na realização das tarefas propostas pelo professor, através do Moodle da disciplina. Essas atividades deverão ser realizadas até a data prevista. Todas as atividades serão propostas, entregues e avaliadas no Moodle da disciplina, onde constarão as instruções a serem seguidas para sua realização. Em caso de dúvidas, os alunos poderão contar com fórum no Moodle da disciplina, troca de mensagens individuais com o professor, e atendimento individualizado do professor, em horário a ser combinado e realizado de forma remota.	
<i>Estratégias didáticas em atividades presenciais</i>	Não serão realizadas atividades presenciais.	
<i>Recursos disponibilizados</i>	As atividades previstas assim como as instruções para sua realização serão disponibilizadas no Moodle do INF. Eventuais componentes externos ao Moodle e necessários para a realização das atividades estarão indicados no próprio Moodle. Também serão disponibilizados no Moodle links para os materiais online e gratuitos, indicados para leitura e estudo, visando auxiliar na realização das atividades propostas.	
<i>Recursos computacionais</i>	Para acompanhar as atividades previstas é necessário ter acesso regular à Internet. As atividades instrucionais síncronas serão disponibilizadas através de um link informado no Moodle, e podem ser acompanhadas através de smartphone ou computador, com microfone e câmera. Para a realização das atividades propostas será necessário ter acesso à computador ou máquina virtual onde possam ser instalados os recursos disponibilizados pelo professor nas aulas práticas (JDK, Eclipse e Astah).	

<b>Informações sobre Direitos Autorais e de Imagem:</b>	<p>Todos os materiais disponibilizados são exclusivamente para fins didáticos, sendo vedada a sua utilização para qualquer outra finalidade, sob as penas legais.</p> <p>Todos os materiais de terceiros que venham a ser utilizados devem ser referenciados, indicando a autoria, sob pena de plágio. A liberdade de escolha de exposição da imagem e da voz não isenta o aluno de realizar as atividades originalmente propostas ou alternativas;</p> <p>Todas as gravações de atividades síncronas devem ser previamente informadas por parte dos professores.</p> <p>Somente poderão ser gravadas pelos alunos as atividades síncronas propostas mediante concordância prévia dos professores e colegas, sob as penas legais.</p> <p>É proibido disponibilizar, por quaisquer meios digitais ou físicos, os dados, a imagem e a voz de colegas e do professor, sem autorização específica para a finalidade pretendida.</p> <p>Os materiais disponibilizados no ambiente virtual possuem licença de uso e distribuição específica, sendo vedada a distribuição do material cuja a licença não permita ou sem a autorização prévia dos professores para o material de sua autoria.</p>
<b>Carga Horária</b> Teórica Prática	<p>44</p> <p>16</p>
<b>Experiências de Aprendizagem</b>	<p>O conteúdo programático será apresentado na forma de slides. Adicionalmente, serão realizadas atividades síncronas coletivas para indicação dos principais aspectos e esclarecimentos de dúvidas. Serão propostas atividades relacionadas com os conteúdos estudados, a serem realizadas de forma assíncrona pelos discentes. Essas atividades serão posteriormente resolvidas através de apresentação assíncrona. Além das atividades semanais, estão previstas encontros síncronos para realização de laboratórios práticos pelos alunos, com a orientação remota do professor. Estas atividades podem ser feitas no horário de aula ou individualmente com base no conteúdo disponibilizado.</p>
<b>Critérios de Avaliação</b>	<p>Ao longo do curso, exercícios serão propostos e resolvidos no prazo especificado para cada exercício. Esses exercícios permitirão ao docente tomar conhecimento do quanto os alunos estão aprendendo.</p> <p>Cada exercício recebe uma nota entre 0 e 10. A média final <math>m</math> é a média aritmética de todas notas. O conceito final é</p> <p>A, caso <math>m \geq 9</math>,</p> <p>B, caso <math>7.5 \leq m &lt; 9</math>, e</p> <p>C, caso <math>6.0 \leq m &lt; 7.5</math>.</p> <p>Para <math>m &lt; 6</math> ver Atividades de Recuperação.</p> <p>De acordo com a Resolução do CEPE sobre o ERE, durante o período em que perdurar o ERE, fica inaplicável a atribuição de conceito FF, prevista no Parágrafo 2º, do Artigo 44, da Resolução nº 11/2013 do CEPE.</p> <p>Para os estudantes matriculados até o final do período e que deixaram de participar da Atividade de Ensino, deverá ser atribuído o registro NI (Não Informado) no campo de conceito do sistema acadêmico.</p> <p>Para os casos previstos no Parágrafo 1º, a justificativa do registro NI deverá conter a referência ao período de excepcionalidade.</p> <p>Os casos de não informação de conceito durante o ERE, deverão ser resolvidos até o fim do segundo período letivo, após o fim da situação emergencial de saúde.</p>
<b>Atividades de Recuperação Previstas</b>	<p>Exercícios não realizados pelos alunos ou resolvidos de forma errada poderão ser refeitos para assegurar o aprendizado e permitir o avanço dos conteúdos.</p> <p>Cada exercício refeito recebe uma nova nota entre 0 e 10. A média final após a recuperação é a média de todas notas (novas, caso houver, senão antigas). O conceito final é obtido como descrito em Critérios de Avaliação.</p>
<b>Bibliografia</b>	<p>Sem alterações</p> <p><b>Básica Essencial</b></p> <p>Daltro José Nunes. Introdução a Abstração de Dados. Porto Alegre: Bookman, 2012. Disponível em versão eletrônica no Sabi+.</p> <p><b>Básica</b></p> <p>Cohen, Bernard; Harwood, William T.; Jackson, Melvyn I.. The specification of complex systems. Wokingham: Addison-Wesley, c1986.</p> <p>Hoare, Charles Antony Richard. Communicating sequential processes. Englewood Cliffs: Prentice-Hall International, c1985. ISBN 0131532715.</p> <p>Milner, Robin. A calculus of communicating systems. Berlin: Springer-Verlag, 1980. ISBN 3540102353.</p> <p>Milner, Robin. Communication and concurrency. New York: Prentice-Hall, c1989. ISBN 0131149849.</p> <p>Watt, David Anthony; Thomas, Muffy. Programming language syntax and semantics. New York: Prentice Hall, c1991. ISBN 0137262663.</p>