

UNIVERSIDADE FEDERAL DE SÃO CARLOS
CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA
DEPARTAMENTO DE COMPUTAÇÃO

PROJETO PEDAGÓGICO DO CURSO DE
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

São Carlos, 10 de outubro de 2025

UNIVERSIDADE FEDERAL DE SÃO CARLOS

Reitora

Prof.ª Dr.ª Ana Beatriz de Oliveira

Pró-Reitor de Graduação

Prof. Dr. Douglas Verrangia Correa da Silva

Diretora do Centro de Ciências Exatas e de Tecnologia

Prof. Dr. Luiz Fernando de Oriani e Paulillo

BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

Coordenação do Bacharelado em Ciência da Computação

Prof.ª Dr.ª Helena de Medeiros Caseli (Coordenadora)

Prof. Dr. Delano Medeiros Beder (Vice-Cordenador)

Secretaria do Curso

Sr.ª Dalila Ariana de Abreu Bernardino

Coordenadora de Estágio

Prof.ª Dr.ª Sandra Abib

Chefe do Departamento de Computação

Prof.ª Dr.ª Marilde Terezinha Prado Santos

Composição do Núcleo Docente Estruturante - NDE

Prof. Dr. Daniel Lucrédio

Prof. Dr. Delano Medeiros Beder

Prof. Dr. Edilson Reis Rodrigues Kato

Prof.ª Dr.ª Helena de Medeiros Caseli (presidente)

Prof. Dr. Hélio Crestana Guardia

Prof. Dr. Mário Cesar San Felice

Prof. Dr. Renato Bueno

Dados de identificação do curso

Campus: São Carlos

Centro: Centro de Ciências Exatas e de Tecnologia

Denominação: Bacharelado em Ciência da Computação

Modalidade: Presencial

Número de Vagas Anuais: 60

Turno de Funcionamento: Integral

Carga Horária Total: 3.240

Regime acadêmico: Semestral

Tempo de Duração do Curso: 8 semestres

Prazo mínimo para integralização curricular: 6 semestres

Prazo máximo para integralização curricular: 14 semestres

Ato legal do primeiro reconhecimento pelo MEC: Parecer 1522/79, em 11 de novembro de 1979

Ano da última reformulação curricular: 2018

Conteúdo

1	Introdução	10
1.1	Objetivos	11
1.2	Organização deste Documento	11
2	Marco Referencial do Curso	13
2.1	Contextualização da Área Ciência da Computação	13
2.2	Décadas de Evolução do Bacharelado em Ciência da Computação	15
3	Marco Conceitual do Curso	20
3.1	Perfis Genéricos da Área da Computação	20
3.2	Perfil do Egresso deste Curso	23
3.3	Aptidões Específicas da Formação	25
3.3.1	Competências para Resolução de Problemas	25
3.3.2	Competências para Compreensão de Conteúdos	26
3.3.3	Competências para Desenvolvimento e Condução de Projetos	26
3.4	Aptidões Gerais da Formação	27
3.4.1	Habilidades Práticas	27
3.4.2	Competências para o Trabalho em Equipe	27
4	Marco Estrutural do Curso	30

4.1	Visão Geral dos Tipos de Atividades Curriculares	30
4.2	Núcleos de Conhecimento	35
4.3	Trilhas Acadêmicas	37
4.4	Matriz Curricular	44
4.5	Ementário	56
4.5.1	Disciplinas do Primeiro Semestre	56
4.5.2	Disciplinas do Segundo Semestre	62
4.5.3	Disciplinas do Terceiro Semestre	70
4.5.4	Disciplinas do Quarto Semestre	78
4.5.5	Disciplinas do Quinto Semestre	85
4.5.6	Disciplinas do Sexto Semestre	98
4.5.7	Disciplinas do Sétimo Semestre	112
4.5.8	Disciplinas do Oitavo Semestre	121
4.6	Dispensas entre Matrizes Curriculares	123
4.7	Plano de migração para novo currículo	125
4.7.1	Plano de migração para alunos enquadrados no 3º período em 2026/1 . .	126
4.7.2	Plano de migração para alunos enquadrados no 5º período em 2026/1 . .	130
5	Integração Ensino, Pesquisa e Extensão	134
5.1	Articulação Ensino, Pesquisa e Extensão	136
5.2	Atividades de Pesquisa	136
5.3	Atividades de Extensão	137
6	Avaliação da Aprendizagem	139
6.1	Princípios gerais de avaliação da aprendizagem	139
6.2	Formas de Avaliação da Aprendizagem	140

7 Avaliação e Gerenciamento do Curso	143
7.1 Composição e Funcionamento do Conselho de Coordenação do Curso	143
7.2 Núcleo Docente Estruturante	145
7.3 O Coordenador de Núcleo de Conhecimento	145
7.4 Administração e Condução do Curso	147
7.5 Processo para Autoavaliação do Curso	147
7.6 Apoio ao discente	148
8 Plano de Implantação	150
8.1 Espaço físico	150
8.2 Corpo docente e técnico	152
Referências Bibliográficas	155
Apêndices	156
I Regulamento do Estágio Curricular Obrigatório e Não obrigatório	157
II Regulamento do Trabalho de Conclusão de Curso	167
III Regulamento de Atividades Curriculares de Extensão	173
IV Regulamento de Atividades Complementares	180
V Regulamento do Projeto Integrador Extensionista	185

Lista de Figuras

3.1	Representação gráfica do perfil do egresso do curso de Ciência da Computação . . .	28
4.1	Trilhas Acadêmicas do Bacharelado em Ciência da Computação	43
4.2	Matriz curricular completa	45
4.3	Matriz curricular completa na visão do estudante	46

Lista de Quadros

2.1	Quantitativo anual do corpo discente nos últimos 7 anos (dados coletados durante o processo de renovação de reconhecimento de curso e são referentes a Novembro/2024).	18
4.1	Quadro de Integralização Curricular	31
4.2	Matriz Curricular do 1º Semestre (390 horas)	48
4.3	Matriz Curricular do 2º Semestre (390 horas)	48
4.4	Matriz Curricular do 3º Semestre (390 horas)	49
4.5	Matriz Curricular do 4º Semestre (360 horas)	49
4.6	Matriz Curricular do 5º Semestre (300 horas)	50
4.7	Matriz Curricular do 6º Semestre (300 horas)	50
4.8	Matriz Curricular do 7º Semestre (240 horas)	51
4.9	Matriz Curricular do 8º Semestre (escolher entre Estágio ou TCC: 360 horas)	51
4.10	Matriz Curricular – Consolidação da Formação (atividades realizadas durante todo o curso: 510 horas)	51
4.11	Matriz Curricular com as Optativas Recomendadas para o 5º Semestre (escolher três disciplinas: 180 h)	52
4.12	Matriz Curricular com as Optativas Recomendadas para o 6º Semestre (escolher três disciplinas: 180 h)	53

4.13 Matriz Curricular com as Optativas Recomendadas para o 7º Semestre (escolher três disciplinas: 180 h)	54
4.14 Eletivas de Formação Multidisciplinar e Humanística	55
4.15 ACIEPEs frequentemente realizadas pelos estudantes do BCC e recomendadas para o curso (ACES do tipo III)	56
4.16 Dispensas entre Matrizes (disciplinas obrigatórias) - Leitura: Disciplina Nova é Dispensada por Disciplina Antiga	123
4.17 Dispensas: Disciplinas Optativas do Semestre 5 que são dispensadas por Antigas	124
4.18 Dispensas: Disciplinas Optativas do Semestre 6 que são dispensadas por Antigas	125
4.19 Dispensas: Disciplinas Optativas do Semestre 7 que são dispensadas por Antigas	125
4.20 Dispensas para alunos no 3º período	126
4.21 3º Semestre - 2026/1 (450 horas)	127
4.22 4º Semestre - 2026/2 (360 horas)	127
4.23 5º Semestre - 2027/1 (300 horas)	128
4.24 6º Semestre - 2027/2 (300 horas)	128
4.25 7º Semestre - 2028/1 (240 horas)	129
4.26 8º Semestre - 2028/2 (escolher entre Estágio ou TCC: 360 horas)	129
4.27 Dispensas para alunos no 5º período	130
4.28 5º Semestre - 2026/1 (420 horas)	131
4.29 6º Semestre - 2026/2 (300 horas)	131
4.30 7º Semestre - 2027/1 (240 horas)	132
4.31 8º Semestre - 2027/2 (escolher entre Estágio ou TCC: 360 horas)	132
8.1 Laboratórios do Departamento de Computação voltados para o ensino da graduação	152
8.2 Corpo docente atuante no curso de Bacharelado em Ciência da Computação	153

8.3	Corpo técnico-administrativo atuante no DC.	154
-----	---	-----

1. Introdução

Nas últimas décadas, a sociedade tem passado por profundas transformações econômicas, sociais e culturais. Vive-se um grande progresso tecnológico que induz mudanças no papel exercido pela computação em praticamente todos os setores da sociedade. Nesse cenário, o ensino em computação cumpre papel fundamental ao mesmo tempo em que é constantemente desafiado a adaptar-se com qualidade e excelência em um mundo com mudanças constantes.

A Ciência da Computação engloba aspectos teóricos e práticos relacionados ao uso de dispositivos computacionais (hardware) para a execução de tarefas descritas na forma de um programa de computador (software). Desse modo, a Ciência da Computação tem um papel fundamental em diversas áreas, como matemática, engenharias, física, química, biologia e medicina, além de ser essencial no desenvolvimento de quase todos os setores industriais contemporâneos. Cabe dizer que ela está no centro de muitos avanços tecnológicos no século XX e início do século XXI, permeando praticamente todas as áreas do conhecimento.

Pode-se considerar o emprego da Ciência da Computação como atividade fim ou meio. Seu emprego como atividade fim visa evoluir a computação em si, promovendo a criação, melhoria e inovação em hardware e software. Já seu emprego como atividade meio consiste na sua utilização para apoiar o desenvolvimento de outras áreas.

Este documento constitui-se no Projeto Pedagógico do Curso (PPC) de Ciência da Computação modalidade Bacharelado (BCC), da Universidade Federal de São Carlos (UFSCar), campus

de São Carlos. O curso visa gerar egressos com sólida formação em computação e, ao mesmo tempo, que possam responder rapidamente às demandas de um mundo em constante mudança. Para tanto, este PPC tem natureza dual contendo: (i) um conjunto de disciplinas fundamentais para qualquer cientista da computação, que são obrigatórias e estão concentradas na primeira metade do curso; (ii) diversas trilhas acadêmicas constituídas por um conjunto complementar de disciplinas, que são optativas e estão concentradas na segunda metade do curso. Assim, garante-se a solidez na formação de nossos egressos e, ao mesmo tempo, a rápida adaptação às mudanças tecnológicas e à abrangência de aplicações da computação às mais diversas áreas.

1.1 Objetivos

O objetivo geral do Bacharelado em Ciência da Computação da UFSCar, campus de São Carlos, é formar profissionais de excelência que empreguem a computação como atividade fim ou meio de modo a contribuir para o crescimento econômico, social e cultural do país. Esses objetivos são alcançados por meio da exposição do estudante a um conjunto de componentes curriculares diversificado, que proporcionam o desenvolvimento de competências e oferecem conhecimentos atualizados do mercado e da área acadêmica.

1.2 Organização deste Documento

Este documento está organizado em 8 capítulos e 5 apêndices. O Capítulo 2 apresenta a discussão sobre o marco referencial do curso; o Capítulo 3 apresenta a discussão sobre o marco conceitual do curso; o Capítulo 4 apresenta os detalhes da constituição do curso e de seus componentes curriculares; o Capítulo 5 trata da relação entre ensino, pesquisa e extensão; o Capítulo 6 aborda as formas de avaliar a aprendizagem dos estudantes; o Capítulo 7 aborda os mecanismos pelos quais os docentes podem avaliar e manter o bom funcionamento do curso ; o Capítulo 8

discute como esta reformulação curricular será colocada em prática. Os Apêndices trazem os regulamentos do Estágio Curricular obrigatório e não obrigatório (Apêndice I), do Trabalho de Conclusão de Curso (Apêndice II), das Atividades Curriculares de Extensão (Apêndice III), das Atividades Complementares (Apêndice IV), e do Projeto Integrador Extensionista (Apêndice V).

2. Marco Referencial do Curso

2.1 Contextualização da Área Ciência da Computação

Já não é mais possível separar a computação da vida em sociedade. Desde algumas décadas atrás, a tecnologia da computação passou a fazer parte de todos os aspectos da vida humana, chegando ao ponto em que se torna muito difícil encontrar lugares onde não existe ao menos um dispositivo computacional oferecendo suporte às atividades humanas. A tendência de que essa simbiose entre humanos e computadores se torne cada vez mais presente já é uma realidade com os diversos aplicativos e dispositivos que usamos em nosso dia a dia.

Além da ampla presença, é importante destacar a profundidade e a complexidade com que a computação penetra na sociedade. Diferentemente da eletricidade, por exemplo, que está também presente mas que se limita à atuação no mundo físico, a computação tem potencial para utilizar e transformar a informação, o que é mais abstrato e que pode ser muito mais impactante para a sociedade.

Nesse cenário, a Ciência da Computação possui papel central. Em muitas atividades, sejam da indústria, comércio, saúde, entre outras, o domínio da computação passou a ser determinante não apenas para a eficiência, mas também para o sucesso das mesmas. Existem inúmeros exemplos de avanços tecnológicos em que a presença de computação tornou possível a realização de atividades que antes eram inviáveis, como: a análise de imensos volumes de dados, que permite a busca de determinada informação em meio a trilhões de páginas de texto, ou de rostos em meio a uma

multidão; o voo de quadricópteros, impossível de ser realizado manualmente por um ser humano; e num futuro muito próximo, a condução de automóveis com uma taxa mínima de acidentes.

Assim sendo, a sociedade está e continuará constantemente interessada em novas formas de utilizar computação em inúmeros setores. Isso torna o cenário extremamente dinâmico, o que constitui um desafio para a universidade. O profissional de computação deve possuir habilidades não apenas para aplicar conceitos e tecnologias já existentes, mas também para pesquisar e desenvolver novas tecnologias. Assim, é fundamental que o seu conhecimento não seja restrito àquele desenvolvido e estabelecido. O profissional de computação deve ser capaz de construir e renovar seu conhecimento continuamente, por vezes revendo conceitos desatualizados. Sob este ponto de vista, o profissional de computação deve ser, antes de tudo e ao longo de sua carreira, um cientista.

A complexidade do desenvolvimento tecnológico em computação também traz um outro aspecto e um novo desafio para a universidade. Em função do grande desenvolvimento da área nas últimas décadas, já não é mais possível que um único profissional consiga atuar em toda e qualquer área em que a computação está envolvida. A computação evoluiu em sub-áreas do conhecimento que exigem grande grau de aprofundamento para que o profissional possa trabalhar de forma eficaz e com a competência necessária para o sucesso em sua atuação.

Por fim, a ubiquidade da computação leva o profissional para dentro de todos os setores da sociedade. Ainda que exista espaço para a atuação em computação de maneira pura, na maioria dos casos o cientista da computação precisará trabalhar de maneira muito próxima a profissionais de diversas áreas do conhecimento. Sob este ponto de vista, portanto, a computação não deixa de ter o caráter de atividade meio.

O papel da universidade, nesse contexto, é o de preparar o cientista da computação para atuar na sociedade, empregando suas habilidades para resolver problemas que promovam o bem estar da sociedade. Dado o cenário apresentado, a universidade deve:

1. fornecer meios para que o futuro cientista alcance o aprendizado do conhecimento central em computação que já se estabeleceu nas últimas décadas;
2. fornecer meios para que o estudante possa se aprofundar em uma determinada sub-área da computação, de forma que ele possa dominar com mais profundidade aquela área de conhecimento;
3. oferecer meios para que o estudante desenvolva habilidades de pesquisa, investigação, desenvolvimento e inovação tecnológica, permitindo que ele possa ir além do conhecimento adquirido e consiga construir seu próprio conhecimento após formado;
4. estimular, sempre que possível, a proximidade dos estudantes com os diversos setores da sociedade, seja trazendo profissionais para palestras e seminários, ou apresentando desafios e projetos que envolvem a aplicação prática da computação nesses setores.

Este PPC faz uma clara distinção entre conhecimento central e certos aprofundamentos tratando-os de forma diferenciada, conforme os itens 1 e 2 apresentados acima. Também tem a premissa de aumentar as ações referentes aos itens 3 e 4, alcançando um balanceamento melhor entre as habilidades que serão cada vez mais necessárias para o cientista da computação.

2.2 Décadas de Evolução do Bacharelado em Ciência da Computação

O Bacharelado em Ciência da Computação da Universidade Federal de São Carlos, campus São Carlos, foi criado em 1973, implantado em 1975 e reconhecido pelo Ministério da Educação – MEC, por meio do Parecer no. 1522/79, em 11 de novembro de 1979. Foi implantado no mesmo ano da Licenciatura em Matemática e do Bacharelado em Ciências Biológicas. Na época, existiam apenas 7 (sete) cursos na Universidade, sendo 4 (quatro) no Centro de Ciências Exatas e de

Tecnologia. Um desses quatro cursos era o de Processamento de Dados, implantado em 1974, que acabou sendo extinto em 1986.

Em 1997, o curso foi submetido a um processo de auto-avaliação dentro do Programa de Avaliação Institucional das Universidades Brasileiras – PAIUB-SESu/MEC, com a participação de seus discentes, docentes, egressos dos últimos cinco anos e funcionários. O processo desenvolveu-se com o objetivo de analisar o curso enquanto unidade organizacional, nos seguintes aspectos: perfil profissional, currículos e programas, condições de funcionamento, desempenho dos docentes e discentes. Ele também permitiu detectar aspectos positivos e negativos do curso, em nível de profundidade bastante significativo. Desde a implantação do curso é realizado um trabalho de acompanhamento que visa avaliar sua estrutura curricular, assim como seguir as exigências impostas pela evolução natural da área de computação no país.

Em 1994 foi realizada uma reformulação do curso, e a renovação do reconhecimento foi homologada pela Comissão de Especialistas e de Verificação (CEEInf) em 30 de outubro de 2001. A avaliação realizada atribuiu ao curso o conceito global A e apontou os pontos fortes e fracos em relação aos seguintes aspectos: corpo docente, plano pedagógico, infra-estrutura e desempenho.

Em 2006, uma outra reformulação do curso foi realizada no sentido de sanar problemas detectados nas avaliações comentadas anteriormente bem como adequar o curso à nova Lei de Diretrizes e Bases da Educação Nacional (Lei no. 9394, de 20 de dezembro de 1996) e às seguintes determinações do Conselho de Ensino, Pesquisa e Extensão da UFSCar: Parecer CEPE no. 776/01, que estabelece o perfil geral do profissional a ser formado na UFSCar e Portaria GR no. 771/04, que dispõe sobre normas e procedimentos referentes às atribuições de currículo, criações, reformulações e adequações curriculares dos cursos de graduação da UFSCar.

Em 2013, novamente novas alterações mais pontuais foram realizadas no sentido da incorporação de disciplinas voltadas ao aprimoramento da capacidade de desenvolvimento de projetos dos estudantes.

Em 2018, a reformulação estabeleceu princípios que são mantidos na proposta aqui apresentada, como: (i) melhor caracterização da Ciência da Computação por meio dos núcleos de conhecimento, que separam o conhecimento central e fundamental da computação e matemática dos aprofundamentos e atualizações das diversas subáreas; (ii) maior flexibilização da formação por meio das linhas de formação (agora renomeadas para trilhas acadêmicas), que agrupam disciplinas optativas para o aprofundamento de uma formação específica dentro da computação; (iii) menor concentração de disciplinas da matemática, as quais são distribuídas ao longo do curso de modo que o estudante curse apenas uma disciplina de matemática em cada semestre; (iv) aperfeiçoamento de competências de desenvolvimento de projetos, com o incentivo à participação em Projetos Integradores Extensionistas e diversas Atividades de Extensão em parceria com empresas e setores da sociedade; (v) limitação da quantidade de disciplinas por semestre, com no máximo seis disciplinas por semestre, permitindo que o estudante tenha tempo para exercitar, refletir e compreender os conceitos apresentados nas aulas; (vi) maior integração entre conteúdos teóricos e práticos nas disciplinas da matriz curricular; (vii) preparação do estudante para o aprendizado contínuo, com o oferecimento de uma base sólida de conhecimentos e práticas de estudo que permitam ao estudante atualizar-se e aprimorar-se ao longo de toda sua carreira profissional; (viii) aprimoramento da consciência do papel do egresso na sociedade, com o oferecimento de um conjunto diversificado de disciplinas optativas do núcleo de humanas, bem como seminários temáticos e a disciplina de Ética em Computação, no sentido de fornecer ao estudante uma consciência sobre seu papel social e ético na sociedade; e (ix) melhoria contínua do curso promovida pela atuação dos coordenadores de núcleo de conhecimento.

Entre 2023 e 2024 o curso passou por um processo de renovação de reconhecimento de curso realizado pelo Ministério da Educação (MEC). Após três fases de avaliação, em três dimensões – organização didático-pedagógica, corpo docente e infraestrutura – o curso obteve o conceito máximo (5) com um índice contínuo 4,92 (em um intervalo que vai de 0 a 5). Os pontos de

melhoria apontados na avaliação de 2023-2024 embasaram decisões adotadas nesta reformulação curricular. Além desses pontos, esta reformulação também levou em consideração as análises dos coordenadores de núcleo de conhecimento realizadas no ano de 2024 a partir das conversas com os docentes que ministram disciplinas para o curso, bem como os resultados das avaliações promovidas pela Comissão Própria de Avaliação (CPA) da UFSCar e do ENADE.

Nesses 50 anos de existência, o BCC já formou mais de 1800 estudantes. A título de ilustração, a Tabela 2.1 traz alguns números do curso referentes aos últimos 7 anos.

Quadro 2.1: Quantitativo anual do corpo discente nos últimos 7 anos (dados coletados durante o processo de renovação de reconhecimento de curso e são referentes a Novembro/2024).

	2018	2019	2020	2021	2022	2023	2024
Ingressantes	73	70	72	72	68	70	72
Matriculados	7	25	40	56	51	59	62
Concluintes	44	38	51	55	30	30	25
Estrangeiros	0	0	0	0	0	1	2
Matriculados em Estágio Supervisionado	39	43	55	63	58	44	63
Matriculados em TCC	5	10	6	3	1	12	16
Participantes em Projetos de Pesquisa	–	13	9	7	17	23	25
Participantes em Projetos de Extensão	7	4	20	32	83	91	–

Esta atual reformulação do curso está em consonância com o estabelecido na Resolução número 5, de 16 de novembro de 2016, que institui as Diretrizes Curriculares Nacionais para os cursos de graduação na área de Computação, abrangendo os cursos de Bacharelado em Ciência da Computação, Bacharelado em Sistemas de Informação, Engenharia de Computação, Bacharelado em Engenharia de Software e de Licenciatura em Computação (MINISTÉRIO DA EDUCAÇÃO - MEC, 16 de Novembro de 2016). Além disso, também está em consonância tanto com o Plano de Desenvolvimento Institucional (PDI) da UFSCar (UNIVERSIDADE FEDERAL DE

SÃO CARLOS - UFSCAR, 2024) quanto com o Regimento Geral dos Cursos de Graduação dessa Universidade (UNIVERSIDADE FEDERAL DE SÃO CARLOS - UFSCAR, Setembro de 2016). Por fim, esta reformulação atualiza o conteúdo do antigo PPC com base nas normativas para inserção curricular da extensão e criação de trilhas acadêmicas nos cursos de graduação da UFSCar.

3. Marco Conceitual do Curso

Este capítulo visa cobrir aspectos relacionados com o perfil do egresso deste curso. Para isso, inicialmente são apresentados os perfis genéricos dos principais cursos da área da computação; Ciência da Computação, Engenharia da Computação, Sistemas de Informação e Engenharia de Software. Logo após, apresenta-se um detalhamento que é específico do egresso do Bacharelado em Ciência da Computação da UFSCar, campus São Carlos.

3.1 Perfis Genéricos da Área da Computação

De acordo com a resolução Nro 5 de 16 de Novembro de 2016 (MINISTÉRIO DA EDUCAÇÃO - MEC, 16 de Novembro de 2016), os egressos dos cursos de computação são diferenciados de acordo com as habilidades e competências que eles devem possuir. Assim, almeja-se que os egressos desses cursos apresentem as seguintes habilidades/competências:

Ciência da Computação:

I - Sólida formação em Ciência da Computação e Matemática que os capacitem a: (i) construir aplicativos de propósito geral, ferramentas e infraestrutura de software de sistemas de computação e de sistemas embarcados, (2) a gerar conhecimento científico e inovação e (3) a estender suas competências à medida que a área se desenvolve;

II - Visão global e interdisciplinar de sistemas com o entendimento de que esta visão transcende os detalhes de implementação dos vários componentes e os conhecimentos dos domínios de aplicação;

III - Conhecimento sobre a estrutura dos sistemas de computação e os processos envolvidos na sua construção e análise;

IV - Domínio dos fundamentos teóricos da área de Computação e como eles influenciam a prática profissional;

V - Capacidade de agir de forma reflexiva na construção de sistemas de computação, compreendendo o seu impacto direto ou indireto sobre as pessoas e a sociedade;

VI - Capacidade de criar soluções, individualmente ou em equipe, para problemas complexos caracterizados por relações entre domínios de conhecimento e de aplicação;

VII - Capacidade de reconhecer o caráter fundamental da inovação e da criatividade com a compreensão das perspectivas de negócios e oportunidades relevantes.

Engenharia da Computação:

I - Sólida formação em Ciência da Computação, Matemática e Eletrônica visando à análise e ao projeto de sistemas de computação, incluindo sistemas voltados à automação e controle de processos industriais e comerciais, sistemas e dispositivos embarcados, sistemas e equipamentos de telecomunicações e equipamentos de instrumentação eletrônica;

II - Conhecimento sobre os direitos e as propriedades intelectuais inerentes à produção e à utilização de sistemas de computação;

III - Capacidade de agir de forma reflexiva na construção de sistemas de computação, compreendendo o seu impacto direto ou indireto sobre as pessoas e a sociedade;

IV - Entendimento sobre o contexto social no qual a Engenharia é praticada, bem como os efeitos dos projetos de Engenharia na sociedade;

V - Capacidade de considerar sobre os aspectos econômicos, financeiros, de gestão e de qualidade, associados a novos produtos e organizações;

VI - Capacidade de reconhecer o caráter fundamental da inovação e da criatividade com a compreensão das perspectivas de negócios e oportunidades relevantes.

Engenharia de Software:

I - Sólida formação em Ciência da Computação, Matemática e Produção, visando a criação de sistemas de software de alta qualidade de maneira sistemática, controlada, eficaz e eficiente que levem em consideração questões éticas, sociais, legais e econômicas;

II - Capacidade de criar, individualmente ou em equipe, soluções para problemas complexos caracterizados por relações entre domínios de conhecimento e de aplicação;

III - Capacidade de agir de forma reflexiva na construção de software, compreendendo o seu impacto direto ou indireto sobre as pessoas e a sociedade;

IV - Entendimento sobre o contexto social no qual a construção de Software é praticada, bem como os efeitos dos projetos de software na sociedade;

V - Capacidade de compreender os aspectos econômicos e financeiros, associados a novos produtos e organizações;

VI - Capacidade de reconhecer o caráter fundamental da inovação e da criatividade com a compreensão das perspectivas de negócios e oportunidades relevantes.

Sistemas de Informação:

I - Sólida formação em Ciência da Computação, Matemática e Administração visando o desenvolvimento e a gestão de soluções baseadas em tecnologia da informação para os processos de negócio das organizações de forma que elas atinjam efetivamente seus objetivos estratégicos de negócio;

II - Capacidade de determinar os requisitos, de desenvolver, de evoluir e de administrar os sistemas de informação das organizações, assegurando que elas tenham as informações e os sistemas de que necessitam para prover suporte a suas operações e obter vantagem competitiva;

III - Capacidade de inovar, planejar e gerenciar a infraestrutura de tecnologia da informação em organizações, bem como desenvolver e evoluir sistemas de informação para uso em processos organizacionais, departamentais e/ou individuais;

IV - Capacidade de escolher e configurar equipamentos, sistemas e programas para a solução de problemas que envolvam a coleta, processamento e disseminação de informações;

V - Entendimento sobre o contexto, envolvendo as implicações organizacionais e sociais, no qual as soluções de sistemas de informação são desenvolvidas e implantadas;

VI - Capacidade de compreender os modelos e as áreas de negócios, atuando como agentes de mudança no contexto organizacional;

VII - Pensamento sistêmico que permita analisar e entender os problemas organizacionais.

3.2 Perfil do Egresso deste Curso

A sólida formação teórica e técnica que este curso oferece, combinada com possibilidades de aprofundamento em diversas áreas, faz com que o egresso deste curso exiba três habilidades/competências que o caracterizam.

A primeira é a capacidade de desenvolver soluções computacionais eficientes para os mais diversos tipos de problemas do cotidiano. Isso o habilita tanto a atuar em empresas nacionais e multinacionais da área de tecnologia, podendo assumir as mais variadas posições de emprego existentes, quanto a seguir uma carreira empreendedora, criando sua própria empresa.

A segunda é a aptidão para seguir carreira científica, realizando mestrado, doutorado, pós-doutorado, e tornando-se pesquisador em universidades e/ou centros de pesquisa nacionais e

internacionais. Essa linha de atuação permite que o egresso possa contribuir para a evolução da própria área de Ciência da Computação.

A terceira é a capacidade de trabalhar éticamente em equipes, sabendo lidar com as diferenças de opiniões e respeitando diferentes pontos de vista, e sempre sabendo colocar suas posições educadamente.

Atualmente, as empresas necessitam de profissionais habilitados para lidar com a grande quantidade de tecnologias que surgem diariamente na área da computação. Dessa forma, o egresso do Bacharelado em Ciência da Computação possui aptidões que lhe permitem atuar nos seguintes setores: i) instituições de ensino ou pesquisa, ii) organizações envolvidas com a produção de software ou hardware, iii) organizações em que software ou hardware é produto de apoio a suas atividades principais e iv) organizações que atuam na área de infraestrutura computacional. Além disso, o egresso está apto a empreender e gerar conhecimento científico e inovação, estendendo suas competências à medida que a área evolui (ZORZO et al., 2017).

Vale ressaltar que as competências e habilidades adquiridas pelo egresso são também adaptáveis a novas demandas que o mercado de trabalho local e regional apresentem. Essa adaptabilidade é obtida por meio do aprendizado de fundamentos e conceitos básicos, que são independentes das tecnologias e métodos atualmente em uso. Desse modo, tem-se que a matriz curricular está organizada de forma a oferecer uma base sólida de conhecimentos teóricos e práticos, articulando disciplinas tradicionais com conteúdos voltados às novas demandas tecnológicas e sociais. Esse perfil reflete as competências necessárias para uma formação sólida e interdisciplinar, englobando habilidades técnicas, científicas e socioemocionais indispensáveis para a atuação em um mercado de trabalho dinâmico e competitivo.

O curso também articula essas competências com as necessidades locais e regionais, promovendo uma formação que valoriza a aplicação prática do conhecimento para resolver problemas específicos do entorno social e econômico. Por meio de projetos de extensão, parcerias com

empresas e iniciativas voltadas ao desenvolvimento da indústria tecnológica regional, os estudantes são preparados para contribuir de maneira significativa com a comunidade. Esse alinhamento fortalece a relevância do curso, assegurando que o egresso esteja apto a atuar em um mercado diversificado e atender às exigências de desenvolvimento regional sustentável.

Além disso, o perfil do egresso é constantemente ampliado para acompanhar as novas demandas do mundo do trabalho, em contínua transformação devido aos avanços tecnológicos e às mudanças sociais. O curso adota práticas pedagógicas inovadoras e incentiva a formação continuada dos estudantes, integrando conteúdos relacionados a áreas emergentes como inteligência artificial, ciência de dados e segurança cibernética. Essa flexibilidade e atualização constantes garantem que os egressos estejam preparados para enfrentar os desafios contemporâneos e contribuir com soluções que impactem positivamente a sociedade. Por fim, vale mencionar que a adoção de metodologias ativas em vários componentes curriculares favorece diferentes perfis de aprendizagem.

A seguir são apresentados os resultados esperados para os processos de aprendizagem e formação proporcionados pelo curso.

3.3 Aptidões Específicas da Formação

3.3.1 Competências para Resolução de Problemas

O egresso do curso de Bacharelado em Ciência da Computação estará apto a:

- Resolver problemas matemáticos e lógicos relacionados à computação;
- Resolver problemas relacionados a aspectos tecnológicos em computação;
- Identificar problemas computacionalmente intratáveis e insolúveis, e possivelmente propor soluções aproximadas;
- Resolver problemas envolvendo soluções computacionais aplicadas a outras áreas do conhecimento humano;
- Identificar questões éticas, sociais e legais na área da computação, e consequentemente buscar soluções não conflitantes com essas questões;

- Analisar criticamente as possíveis soluções computacionais em relação a seus impactos na sociedade e no meio ambiente, em consonância com os objetivos de desenvolvimento sustentável (ODS) definidos pela ONU.

3.3.2 Competências para Compreensão de Conteúdos

Usando o conhecimento adquirido durante o curso de Bacharelado em Ciência da Computação o egresso será capaz de:

- Compreender e aplicar uma ampla gama de princípios e ferramentas disponíveis para o cientista da computação, conforme abordado pela matriz curricular;
- Compreender textos técnicos, inclusive aqueles escritos em língua inglesa;
- Aplicar seu senso crítico de forma abrangente em diversos campos mais aprofundados da ciência da computação, permitindo o eventual aprimoramento desses campos em estudos de pós-graduação;
- Aplicar seus conhecimentos sobre metodologias de pesquisa, permitindo o engajamento em atividades de pesquisa científica dentro e fora da academia;
- Reconhecer e valorizar atributos de qualidade, responsabilidades profissionais e éticas do profissional de computação;
- Estabelecer a relação entre os conteúdos vistos nas disciplinas e suas aplicações em problemas reais, visando um diálogo construtivo e transformador com os demais setores da sociedade.

3.3.3 Competências para Desenvolvimento e Condução de Projetos

No que diz respeito ao projeto de sistemas computacionais, o egresso do curso de Bacharelado em Ciência da Computação será capaz de:

- Elaborar e conduzir projetos acadêmicos e profissionais na área de ciência da computação;
- Projetar e programar sistemas de software e hardware;
- Apresentar soluções computacionais a uma equipe, arguindo sobre as vantagens e desvantagens da solução;
- Coordenar o processo de desenvolvimento de software, utilizando metodologias adequadas;
- Construir um plano de negócios na área de computação;
- Gerenciar equipes de forma a obter bons níveis de produtividade e também possuir comportamento adequado ao participar de equipes, cooperando com os demais membros;

- Envolver-se proativamente em iniciativas que expressam o compromisso social, em consonância com as políticas ligadas à questões ambientais, étnico-raciais, direitos humanos, entre outras, considerando a interprofissionalidade e interdisciplinaridade .

3.4 Aptidões Gerais da Formação

3.4.1 Habilidades Práticas

No que diz respeito às habilidades práticas, o egresso do curso de Bacharelado em Ciência da Computação será capaz de:

- Planejar e realizar projetos individuais de pequeno ou grande porte;
- Elaborar relatórios técnicos e apresentá-los com competência, em formato escrito ou oral;
- Conduzir apresentações técnicas adequadas para o tempo, lugar e público;
- Usar a literatura científica de forma eficaz, bem como os recursos disponíveis na Web;
- Utilizar ferramentas computacionais apropriadas para dar suporte à condução de projetos;
- Aplicar conhecimentos de computação em ambiente comercial, industrial ou de pesquisa;
- Demonstrar iniciativa, inovação e autogestão na condução de projetos;
- Integrar as competências previamente adquiridas e aplicá-las a novas situações;
- Identificar possíveis contribuições para enfrentamento de questões da sociedade, em contexto local ou global, em consonância com os ODS.

3.4.2 Competências para o Trabalho em Equipe

No que diz respeito ao trabalho em equipe, o egresso do Bacharelado em Ciência da Computação será capaz de:

- Demonstrar capacidade de trabalho em grupo;
- Trabalhar eficazmente com e para outras pessoas;
- Liderar equipes;
- Comunicar-se de forma efetiva com pessoas da área de computação e de outras áreas, em qualquer nível hierárquico;
- Encontrar o equilíbrio entre a autossuficiência e a busca por ajuda;
- Realizar funções técnicas e administrativas dentro de uma equipe de trabalho;

- Mostrar responsabilidade para o cumprimento de prazos e metas de custos;
- Desenvolver planos de carreira e objetivos pessoais;
- Comportar-se de modo ético, crítico e responsável em relação à equipe, ao projeto e aos demais setores e pessoas envolvidos, respeitando as especificidades de cada um.

A Figura 3.1 traz uma ilustração gráfica do perfil do egresso do Bacharelado em Ciência da Computação. A figura destaca três grandes conceitos que são base para o curso: Núcleos de Conhecimento, Trilhas Acadêmicas e Atividades complementares (algumas delas de caráter extensionista).

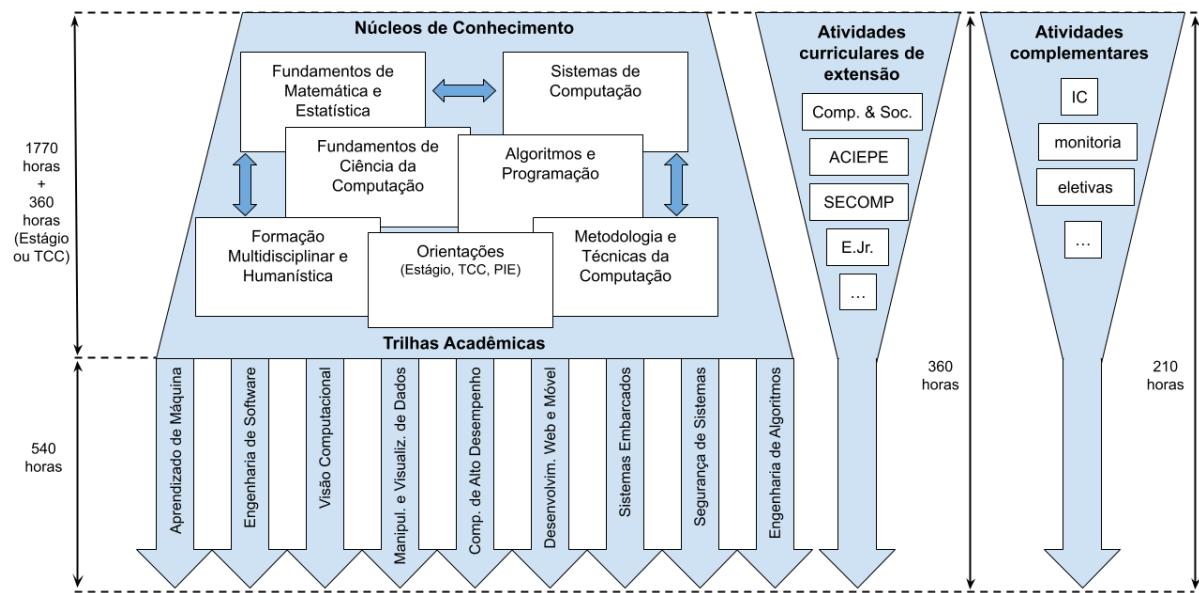


Figura 3.1: Representação gráfica do perfil do egresso do curso de Ciência da Computação

Os Núcleos de Conhecimento são componentes curriculares abstratos que agrupam disciplinas relacionadas. Este curso está dividido em 7 Núcleos de Conhecimento, como pode ser visto na Figura 3.1. A sobreposição existente entre alguns Núcleos de Conhecimento revela a multidisciplinaridade existente. Em nível mais granular, essa multidisciplinaridade é implementada por meio de projetos comuns entre disciplinas e outros mecanismos pedagógicos adequados.

A figura também ressalta o conceito de Trilhas Acadêmicas, mostrando que assim que o estudante foi capacitado na área fundamental e base do curso, ele poderá, a critério próprio, se aprofundar em determinados temas. Este curso oferece várias Trilhas Acadêmicas.

Em paralelo a este cenário pedagógico que mescla capacitação em áreas fundamentais e flexibilização para formações mais específicas, encontram-se as atividades complementares, representadas pelo funil do lado direito da figura, algumas delas com caráter extensionista (atividades curriculares de extensão). Essas atividades visam complementar a formação que o estudante recebe em sala de aula, expondo-o a situações e experiências que fortalecem o desenvolvimento de competências e habilidades diversificadas. Essas experiências tornam o egresso do curso um profissional totalmente preparado para o ambiente real de trabalho, seja na indústria ou na academia.

Detalhes mais específicos sobre os Núcleos de Conhecimento, Trilhas Acadêmicas e Atividades encontram-se no decorrer deste documento.

4. Marco Estrutural do Curso

Este capítulo apresenta detalhes da estrutura completa do Bacharelado em Ciência da Computação da Universidade Federal de São Carlos, campus São Carlos. Inclui visão geral dos tipos de atividades curriculares existentes, detalhes de integralização curricular, apresentação dos Núcleos de Conhecimento do curso, das Trilhas Acadêmicas, a matriz curricular e demais aspectos relacionados.

4.1 Visão Geral dos Tipos de Atividades Curriculares

O marco estrutural do Bacharelado em Ciência da Computação da UFSCar, campus de São Carlos, é constituído pelos seguintes tipos de componentes curriculares: *disciplinas obrigatórias*; *disciplinas optativas profissionalizantes*; *atividades curriculares de extensão*; *atividades complementares*; e *trabalho de conclusão de curso e estágio*. A carga horária em cada um desses tipos de componentes curriculares é apresentada no Quadro 4.1. Note-se que o curso todo exige o cumprimento de 3.240 horas em atividades curriculares, o que atende às diretrizes curriculares nacionais (MINISTÉRIO DA EDUCAÇÃO - MEC, 16 de Novembro de 2016).

As disciplinas obrigatórias visam desenvolver competências e habilidades que são essenciais na formação de um bacharel em Ciência da Computação. Essas disciplinas visam atender às

Quadro 4.1: Quadro de Integralização Curricular

Componentes Curriculares	Carga Horária	
	Teórica/Prática	Extensão
Disciplinas Obrigatórias	1.770	60
Disciplinas Optativas Profissionalizantes	540	0
Atividades Curriculares de Extensão	0	300
Atividades Complementares	210	0
Estágio ou TCC (Trabalho de Conclusão do Curso)	360	0
Total de Horas para Integralização	2.880	360
		3.240 Horas

diretrizes curriculares nacionais determinadas pelo ministério da educação (MINISTÉRIO DA EDUCAÇÃO - MEC, 16 de Novembro de 2016).

As disciplinas optativas profissionalizantes fornecem aprofundamento em determinadas áreas da computação e também colaboram para tornar o curso mais flexível, permitindo que os estudantes possam adaptar o curso a demandas específicas, sejam elas pessoais, da academia, ou de mercado. O oferecimento de um conjunto significativo de disciplinas optativas profissionalizantes foi um dos norteadores da reformulação de 2018 e é seguido nesta reformulação curricular. Assim, uma parcela significativa da carga horária em disciplinas que é exigida para a integralização curricular pode ser adaptada pelo próprio estudante.

As Atividades Curriculares de Extensão (ACEs) podem ser de três tipos: (I) Atividades Curriculares Obrigatórias, Optativas ou Eletivas com carga horária integral ou parcial voltada à abordagem extensionista; (II) Atividades Curriculares de Integração entre Ensino, Pesquisa e Extensão (ACIEPEs) previstas na matriz curricular; e (III) Atividades Complementares de Extensão: Ações de extensão, com ou sem bolsa, com aprovação registrada na Pró-Reitoria de Extensão nas modalidades de projetos, cursos, oficinas, eventos, prestação de serviços e ACIEPEs não previstas na matriz curricular. Este curso de Ciência da Computação prevê uma disciplina obrigatória extensionista (ACE do tipo I) e diversas oportunidades de atividades complementares

de extensão (ACEs do tipo III) para totalizar a carga horária mínima de 10% em atividades de extensão. A regulamentação para as ACEs no curso é apresentada no Apêndice III.

As atividades complementares são atividades curriculares que não estão compreendidas no desenvolvimento regular das disciplinas do curso. São atividades de caráter acadêmico, científico, social e cultural realizadas pelo estudante ao longo de seu curso de graduação, que contribuem para o enriquecimento científico, profissional e cultural e para o desenvolvimento de valores e hábitos de colaboração e de trabalho em equipe. Este curso de Ciência da Computação prevê uma lista extensa e muitas oportunidades de atividades complementares que são oferecidas para os estudantes como: iniciações científicas para engajamento do estudante em atividades de pesquisa; atuação como monitor em disciplinas; participação como ouvinte em eventos científicos e acadêmicos etc. O Apêndice IV traz o regulamento das Atividades Complementares para este curso.

A diversidade das atividades complementares previstas para o BCC é um dos pontos fortes do curso, permitindo aos estudantes escolherem entre uma ampla gama de opções que atendam às suas necessidades e interesses individuais. São oferecidas oportunidades que abrangem desde a participação em eventos acadêmicos e técnicos até o envolvimento em projetos de extensão e programas de iniciação científica. A aderência dessas atividades à formação geral e específica dos estudantes é assegurada por mecanismos de validação que garantem sua relevância e contribuição para o desenvolvimento das competências desejadas para o egresso. Essas atividades desempenham um papel estratégico na formação integral do egresso, promovendo não apenas sua qualificação técnica, mas também habilidades interpessoais e uma visão ampliada sobre o mercado e a sociedade.

Vale ressaltar, também, que cada atividade possui particularidades em sua execução. Por exemplo, iniciações científicas, participações nos grupos PET e projetos integradores são atividades que possuem a orientação de professores do departamento. Importante observar que as atividades

podem tanto possuir caráter único como múltiplo. Por exemplo, atuações como monitor de disciplinas (monitorias) têm apenas caráter de ensino, enquanto que a participação em grupos PET possui caráter de ensino, pesquisa e extensão. Uma atividade que é bastante particular deste curso de Ciência da Computação são os Projetos Integradores Extensionistas (PIEs). Esses projetos envolvem grupos de alunos e têm como objetivo o desenvolvimento de um produto de software que envolva conceitos de várias disciplinas do curso. Detalhes sobre os PIEs podem ser obtidos no Apêndice V.

O Trabalho de Conclusão de Curso (TCC) é um componente curricular que constitui-se de um trabalho acadêmico sobre um determinado tema relacionado ao Bacharelado em Ciência da Computação. O TCC é desenvolvido sob a supervisão de um docente e formalizado como uma monografia. O objetivo é expor o estudante a temas recentes e que extrapolam os limites dos conteúdos transmitidos durante o curso. No BCC, o estudante tem a liberdade de optar pela realização do TCC ou pela realização do Estágio Obrigatório na finalização de seu curso, isto é, não é obrigatória a realização dos dois. Também vale ressaltar que o TCC pode ser substituído pela realização de um ano de Iniciação Científica. Mais detalhes podem ser obtidos no Apêndice II, que traz a regulamentação do Trabalho de Conclusão de Curso.

O Estágio é uma atividade diferenciada, que é desenvolvida pelo estudante dentro de um ambiente real de trabalho (empresas) e que é supervisionada por um docente do curso e por um supervisor dentro da empresa. O Estágio é previsto para o último semestre do curso e seu objetivo é preparar o estudante para o exercício profissional. Segundo relatos dos próprios estudantes, o contato com problemas reais das empresas e a necessidade de propor soluções computacionais para eles os ajudam a desenvolver competências que complementam sua formação, como a necessidade de serem protagonistas colocando em prática, de modo efetivo, os conceitos vistos em sala de aula. Nesse sentido, a formação forte na fundamentação computacional oferecida pelo BCC (p. ex., Sistemas Operacionais, Banco de Dados, Redes de Computadores, Programação Orientada

a Objetos) é frequentemente apontada pelos supervisores dos estagiários como essencial para prepará-los para exercerem suas atividades de estágio com segurança e qualidade. O regulamento do Estágio faz parte deste projeto pedagógico de curso e está presente no Apêndice I.

De acordo com (MINISTÉRIO DA EDUCAÇÃO - MEC, 16 de Novembro de 2016), o curso deve assegurar na formação do egresso o conhecimento das questões sociais, profissionais, legais, éticas, políticas e humanísticas. Pelo Regimento Geral dos Cursos de Graduação (UNIVERSIDADE FEDERAL DE SÃO CARLOS - UFSCAR, Setembro de 2016) e o Plano de desenvolvimento institucional (UNIVERSIDADE FEDERAL DE SÃO CARLOS - UFSCAR, 2024) também deve-se assegurar o conhecimento das temáticas relacionadas à História e Cultura Afro-Brasileira e Indígena, Língua Brasileira de Sinais (LIBRAS), Educação em Direitos Humanos e Educação Ambiental. No BCC, esses conteúdos são trabalhados de forma transversal nas disciplinas, nas atividades complementares e de extensão que os estudantes devem realizar, bem como na disciplina obrigatória extensionista e eletivas de formação multidisciplinar e humanística listadas no Quadro 4.14.

Vale ressaltar que a estrutura curricular mais flexível do BCC, em que uma parte importante da formação do estudante se dá por meio da sua escolha dentre um conjunto de disciplinas optativas, traz diversos benefícios. Com essa flexibilidade, busca-se não apenas uma estrutura que possa ser facilmente ajustável ao progresso do conhecimento científico e tecnológico, mas também que reforce o senso de responsabilidade e envolvimento do estudante, já que este passa a ser ator central da sua própria formação. Além disso, o curso implementa práticas emergentes na computação, incorporando novas metodologias e tecnologias à formação acadêmica. A adoção de aprendizagem baseada em projetos, metodologias ativas e o incentivo à participação em atividades de pesquisa e extensão asseguram que os estudantes estejam na vanguarda das inovações. Dessa forma, os meios para se alcançar os objetivos do curso são continuamente atualizados, garantindo um perfil do egresso compatível com as demandas contemporâneas e as transformações no campo da computação.

Por fim, vale destacar que a carga horária das atividades curriculares é determinada de acordo com a complexidade dos conteúdos, garantindo aos estudantes tempo suficiente para se aprofundarem nos temas e desenvolverem as competências desejadas. Além disso, este PPC não prevê a possibilidade de oferta parcial da carga horária do curso, ou de atividade curricular, na modalidade a distância.

4.2 Núcleos de Conhecimento

O curso é organizado em Núcleos de Conhecimento para facilitar seu gerenciamento e fornecer uma visão mais macro da estrutura curricular. Cada núcleo de conhecimento engloba um conjunto de disciplinas relacionadas. Essa divisão facilita a administração do curso para manutenção de sua excelência, pois existe a figura do coordenador de núcleo, que é um professor do curso especialista nas disciplinas de um determinado núcleo, podendo reger adequadamente os conteúdos administrados nessas disciplinas.

Para este curso foram definidos sete Núcleos de Conhecimento: (1) Fundamentos de Matemática e Estatística, (2) Fundamentos de Ciência da Computação, (3) Algoritmos e Programação, (4) Metodologia e Técnicas da Computação, (5) Sistemas de Computação, (6) Formação Multidisciplinar e Humanística e (7) Orientações.

No Núcleo de Fundamentos de Matemática e Estatística encontram-se disciplinas como Cálculo Diferencial e Integral (Cálculo 1), Geometria Analítica, Álgebra Linear e Estatística Básica. Esse núcleo oferece aos estudantes os conhecimentos de base da matemática e estatística úteis para diversas disciplinas do curso.

No Núcleo de Fundamentos de Ciência da Computação encontram-se as disciplinas de Matemática Discreta, Lógica Matemática, Teoria da Computação e Compiladores. Esse núcleo capacita o estudante em aspectos teóricos essenciais à Ciência da Computação, e que fornecem

um embasamento mais profundo do que aquele encontrado em cursos voltados especificamente à programação de computadores.

O Núcleo de Algoritmos e Programação tem como foco o ensino do desenvolvimento de algoritmos e programas de computador, essencial para qualquer atividade ligada à computação. Esse núcleo visa desenvolver habilidades e competências relacionadas à construção de sistemas computacionais como atividade fim ou atividade meio. O objetivo das disciplinas deste núcleo de conhecimento é capacitar os estudantes na programação de computadores e na resolução de problemas computacionais, fornecendo a eles as técnicas necessárias para o aproveitamento pleno das demais disciplinas do curso que tratam da computação. Exemplos de disciplinas desse núcleo são: Introdução ao Pensamento Algorítmico, Construção de Algoritmos e Programação, Algoritmos e Estrutura de Dados 1 e 2, Programação Orientada a Objetos e Projeto e Análise de Algoritmos.

O Núcleo de Metodologias e Técnicas da Computação traz uma série de tecnologias e metodologias computacionais como Engenharia de Software, Inteligência Artificial, Interação Humano-Computador e Banco de Dados. Esse núcleo visa desenvolver habilidades e competências relacionadas ao projeto e ao desenvolvimento de software.

O Núcleo de Sistemas de Computação apresenta soluções que são, em sua maioria, base para o funcionamento de software de mais alto nível, como Sistemas Operacionais, Arquitetura e Organização de Computadores e Redes de Computadores. Esse núcleo trata de uma série de tópicos relacionados à construção de sistemas de computação, incluindo aspectos de hardware e software.

O Núcleo de Formação Multidisciplinar e Humanística complementa a formação do estudante com a apresentação de conteúdos interdisciplinares e da ciências humanas. Esse núcleo visa desenvolver habilidades e valores relacionados à atuação do egresso na sociedade com o intuito de formar profissionais conscientes de seus papéis ético, humano e social. Também tem como

objetivo permitir que o estudante diversifique e complemente sua formação. Nesse núcleo estão a disciplinas obrigatórias Computação e Sociedade e Metodologia Científica, sendo complementado pelas diversas disciplinas eletivas listadas no Quadro 4.14.

No Núcleo de Orientações os estudantes também são envolvidos em atividades que demandam supervisão de docentes. Nesse conjunto destacam-se as Iniciações Científicas, o Trabalho de Conclusão de Curso, o Estágio e o Projeto Integrador Extensionista. O objetivo desse núcleo é desenvolver habilidades e competências tanto de atuação do estudante no mercado de trabalho quanto seu desenvolvimento para atuação científica em pesquisas relacionadas com computação.

4.3 Trilhas Acadêmicas

As Trilhas Acadêmicas deste curso foram definidas com base na Resolução CoG 474/2024, que estabelece as diretrizes para a criação das trilhas acadêmicas nos cursos de graduação da UFSCar. Neste sentido, são definidas 9 (nove) trilhas acadêmicas seguindo os princípios elencados na resolução citada: (i) ampliação, flexibilização e enriquecimento da formação; (ii) educação humanizadora e emancipatória; (iii) interdisciplinariedade e interprofissionalidade; e (iv) ODS – Objetivos de Desenvolvimento Sustentável da ONU.

A definição de tantas Trilhas Acadêmicas só é possível graças à quantidade de disciplinas optativas oferecidas. As Trilhas Acadêmicas são constituídas por conjuntos de componentes curriculares, totalizando, no mínimo, 240h cursadas por estudantes de graduação, que serão certificadas ao final do curso. As Trilhas Acadêmicas são opcionais, isto é, o estudante pode optar por não fazer trilhas e seguir para uma formação mais generalista.

O curso oferece 9 (nove) Trilhas Acadêmicas, que são detalhadas a seguir. Todas as Trilhas se enquadram na categoria I: “Trilha interna ao curso: contendo componentes da matriz curricular de caráter diversos, como obrigatórios, optativos e ACIEPEs, sendo esta categoria aderente

a um curso específico devido às questões de prioridades e requisitos de inscrição previstos no PPC". Os Quadros a seguir apresentam as trilhas acadêmicas com a carga horária, a descrição, os componentes curriculares e os Objetivos de Desenvolvimento Sustentável (ODS).

Aprendizado de Máquina	
Nome:	Aprendizado de Máquina
Categoria:	I
Curso de oferta:	BCC
Carga horária mínima:	240h
Carga horária máxima:	240h
Descrição:	A trilha de Aprendizado de Máquina visa ampliar, flexibilizar e enriquecer a formação do estudante abordando temas vinculados ao desenvolvimento de sistemas inteligentes. Ao concluir esta trilha, os estudantes estarão aptos a propor soluções computacionais para diversos problemas da sociedade utilizando modelos de aprendizado de máquina e levando em consideração os impactos dessas soluções na humanidade.
Componentes curriculares:	Inteligência Artificial Aprendizado de Máquina 1 Aprendizado de Máquina 2 Processamento de Linguagem Natural
ODS:	ODS9: Indústria, inovação e infraestrutura ODS4: Educação de qualidade ODS8: Trabalho decente e crescimento econômico
Engenharia de Software	
Nome:	Engenharia de Software
Categoria:	I
Curso de oferta:	BCC
Carga horária mínima:	240h
Carga horária máxima:	300h
Descrição:	A trilha de Engenharia de Software visa ampliar, flexibilizar e enriquecer a formação do estudante abordando temas vinculados ao desenvolvimento de sistemas de software complexos. Ao concluir esta trilha, os estudantes estarão aptos a desenvolver, reutilizar, testar, validar e gerenciar projetos de sistemas de software.
Componentes curriculares:	Engenharia de Software 1 Engenharia de Software 2 Programação Orientada a Objetos Avançada Arquitetura de Software e Padrões DevOps
ODS:	ODS9: Indústria, inovação e infraestrutura ODS4: Educação de qualidade ODS8: Trabalho decente e crescimento econômico

Visão Computacional	
Nome:	Visão Computacional
Categoria:	I
Curso de oferta:	BCC
Carga horária mínima:	330h
Carga horária máxima:	330h
Descrição:	<p>A trilha de Visão Computacional visa ampliar, flexibilizar e enriquecer a formação do estudante abordando temas vinculados ao desenvolvimento de sistemas artificiais que automaticamente extraem informações de imagens (e vídeos) digitais e as usam para a detecção, rastreamento e classificação de objetos nas imagens. Aplicações de Visão Computacional estão presentes em diversos cenários, como reconhecimento ótico de caracteres, inspeção de peças em indústria, construção de modelos 3D em fotometria, imagens médicas, segurança automobilística, robótica, biometria (reconhecimento de íris e impressão digital), câmeras de segurança, entre outras. Ao concluir esta trilha, os estudantes estarão aptos a propor soluções computacionais utilizando as principais técnicas de visão computacional.</p>
Componentes curriculares:	<p>Álgebra Linear Processamento de Sinais Digitais Processamento Digital de Imagens Processamento Digital de Imagens 3D e Vídeos Visão Computacional</p>
ODS:	<p>ODS9: Indústria, inovação e infraestrutura ODS4: Educação de qualidade ODS8: Trabalho decente e crescimento econômico</p>

Manipulação e Visualização de Dados	
Nome:	Manipulação e Visualização de Dados
Categoria:	I
Curso de oferta:	BCC
Carga horária mínima:	240h
Carga horária máxima:	240h
Descrição:	<p>A trilha de Manipulação e Visualização de Dados visa ampliar, flexibilizar e enriquecer a formação do estudante abordando temas vinculados ao processamento inteligente e eficiente dos dados. Ao concluir esta trilha, os estudantes estarão aptos a propor soluções computacionais para problemas que envolvam armazenamento, manipulação e visualização de dados, inclusive em larga escala.</p>
Componentes curriculares:	<p>Banco de Dados Projeto e Implementação de Banco de Dados Banco de Dados para Ciência de Dados Processamento e Visualização de Dados</p>
ODS:	<p>ODS9: Indústria, inovação e infraestrutura ODS8: Trabalho decente e crescimento econômico ODS4: Educação de qualidade</p>

Computação de Alto Desempenho	
Nome:	Computação de Alto Desempenho
Categoria:	I
Curso de oferta:	BCC
Carga horária mínima:	330h
Carga horária máxima:	330h
Descrição:	<p>A trilha de Computação de Alto Desempenho visa ampliar, flexibilizar e enriquecer a formação do estudante abordando temas vinculados ao desenvolvimento de sistemas computacionais com múltiplos elementos de processamento. A computação de alto desempenho está comumente associada ao processamento paralelo, que pode ser explorado via instruções vetorizadas, combinando o uso de processadores tradicionais com dispositivos de processamento especializados, bem como às tecnologias de interligação em rede permitem que múltiplos sistemas computacionais cooperem na execução de programas. Ao concluir esta trilha, os estudantes estarão aptos a trabalhar com sistemas de alto desempenho, explorando aspectos que vão desde a arquitetura de sistemas computacionais paralelos, passando por recursos oferecidos pelos sistemas operacionais para a criação de tarefas locais e remotas, além de técnicas de programação para particionamento e sincronização de atividades em sistemas computacionais distribuídos.</p>
Componentes curriculares:	<p>Sistemas Operacionais Sistemas Distribuídos Programação Paralela e Distribuída Arquiteturas de Alto Desempenho Processamento de Dados em Escala</p>
ODS:	<p>ODS9: Indústria, inovação e infraestrutura ODS4: Educação de qualidade ODS8: Trabalho decente e crescimento econômico</p>

Desenvolvimento Web e Móvel	
Nome:	Desenvolvimento Web e Móvel
Categoria:	I
Curso de oferta:	BCC
Carga horária mínima:	240h
Carga horária máxima:	240h
Descrição:	A trilha de Desenvolvimento Web e Móvel visa ampliar, flexibilizar e enriquecer a formação do estudante abordando temas vinculados ao desenvolvimento de interface web ou para dispositivos móveis. Por mais que existam diferentes capacidades e especialidades sendo requeridas no mercado de trabalho, sem dúvida a demanda por desenvolvedores "full-stack", que engloba o desenvolvimento web e móvel, é a que mais fortemente se apresenta para o profissional. Ao concluir esta trilha, os estudantes estarão aptos a projetar qualquer tipo de aplicação web e móvel, cobrindo os principais conceitos fundamentais, e implementar aplicações para diferentes domínios.
Componentes curriculares:	Interação Humano-Computador Desenvolvimento de Software para Web 1 Desenvolvimento de Software para Web 2 Desenvolvimento Móvel
ODS:	ODS9: Indústria, inovação e infraestrutura ODS4: Educação de qualidade ODS8: Trabalho decente e crescimento econômico

Sistemas Embarcados	
Nome:	Sistemas Embarcados
Categoria:	I
Curso de oferta:	BCC
Carga horária mínima:	270h
Carga horária máxima:	270h
Descrição:	A trilha de Sistemas Embarcados visa ampliar, flexibilizar e enriquecer a formação do estudante abordando temas vinculados ao desenvolvimento de dispositivos eletrônicos para serem incorporados em uma grande variedade de produtos. Ao concluir esta trilha, os estudantes estarão aptos a aplicar conhecimentos avançados de arquitetura de computadores e técnicas modernas de projetos de sistemas digitais voltadas ao desenvolvimento de sistemas dedicados que possam ser implementados como SoC (System-on-Chip) e suas aplicações, como por exemplo na Internet das Coisas "IoT".
Componentes curriculares:	Arquitetura e Organização de Computadores 1 Arquitetura e Organização de Computadores 2 Sistemas Digitais Projeto de Sistemas Computacionais Embarcados
ODS:	ODS9: Indústria, inovação e infraestrutura ODS8: Trabalho decente e crescimento econômico ODS4: Educação de qualidade

Segurança de Sistemas	
Nome:	Segurança de Sistemas
Categoria:	I
Curso de oferta:	BCC
Carga horária mínima:	300h
Carga horária máxima:	300h
Descrição:	A trilha de Segurança de Sistemas visa ampliar, flexibilizar e enriquecer a formação do estudante abordando temas vinculados ao desenvolvimento de sistemas computacionais seguros, ou seja, que estejam menos suscetíveis a ataques e interferências indevidas. Ao concluir esta trilha, os estudantes estarão aptos a desenvolver sistemas que sigam os princípios fundamentais de segurança computacional.
Componentes curriculares:	Redes de Computadores Segurança Cibernética Tecnologia de Comunicação Projeto de Sistemas Computacionais Embarcados Arquiteturas de Alto Desempenho
ODS:	ODS9: Indústria, inovação e infraestrutura ODS8: Trabalho decente e crescimento econômico ODS4: Educação de qualidade

Engenharia de Algoritmos	
Nome:	Engenharia de Algoritmos
Categoria:	I
Curso de oferta:	BCC
Carga horária mínima:	240h
Carga horária máxima:	240h
Descrição:	A trilha de Engenharia de Algoritmos visa ampliar, flexibilizar e enriquecer a formação do estudante abordando temas vinculados ao desenvolvimento de algoritmos. Problemas computacionais complexos e desafiadores surgem nas mais variadas áreas. Pode ser muito difícil, ou mesmo impossível, encontrar algoritmos eficientes que sempre encontram a solução ótima para estes problemas. Para atacá-los é importante usar abordagens avançadas e complementares que permitam o projeto e análise de algoritmos exatos, heurísticos e aproximados. Ao concluir esta trilha, os estudantes estarão aptos a usar várias técnicas que permitem resolver problemas computacionais complexos, tipicamente NP-Difíceis.
Componentes curriculares:	Algoritmos e Estruturas de Dados 2 Projeto e Análise de Algoritmos Projeto e Análise de Algoritmos 2 Otimização Matemática
ODS:	ODS9: Indústria, inovação e infraestrutura ODS4: Educação de qualidade ODS8: Trabalho decente e crescimento econômico

A Figura 4.1 traz a ilustração das nove Trilhas Acadêmicas do curso e seus componentes curriculares. Como a realização das trilhas é opcional para os estudantes, o total de horas por semestre não é apresentado na figura.

	Aprendizado de Máquina	Engenharia de Software	Visão Computacional	Manipulação e Visualização de Dados	Computação de Alto Desempenho	Desenvolvimento Web e Móvel	Sistemas Embarcados	Segurança de Sistemas	Engenharia de Algoritmos
Sem. 1									
Sem. 2									
Sem. 3									
Sem. 4	Inteligência Artificial 30 30	Engenharia de Software 1 60	Álgebra Linear 1 45 15	Banco de Dados 30 30	Sistemas Operacionais 60 30	Interação Humano-Computador 30 30	Arquitetura e Organização de Computadores 1 60 30		Algoritmos e Estruturas de Dados 2 60
Sem. 5	Aprendizado de Máquina 1 60	Engenharia de Software 2 60	Programação Orientada a Objetos Avançada 30 30	Processamento de Sinais Digitais 60 30	Processamento Digital de Imagens 30 30	Projeto e Implementação de Banco de Dados 30 30	Sistemas Distribuídos 60	Desenvolvimento de Software para Web 1 60	Projeto e Análise de Algoritmos 60
Sem. 6	Aprendizado de Máquina 2 60	Arquitetura de Software e Padrões 60	Processamento Digital de Imagens 3D e Vídeos 30 30	Visão Computacional 30 30	Banco de Dados para Ciência de Dados 30 30	Processamento e Visualização de Dados 30 30	Programação Paralela e Distribuída 30 30	Desenvolvimento de Software para Web 2 60	Arquitetura e Organização de Computadores 2 30 30
Sem. 7	Processamento de Linguagem Natural 60	DevOps 60					Arquiteturas de Alto Desempenho 30 30	Processamento de Dados em Escala 30 30	Sistemas Digitais 30 30
Sem. 8							Desenvolvimento Móvel 60	Projeto de Sistemas Computacionais Embarcados 60	Segurança Cibernética 30 30
							Tecnologia de Comunicação 30 30	Projeto de Sistemas Computacionais Embarcados 60	Arquiteturas de Alto Desempenho 30 30

Figura 4.1: Trilhas Acadêmicas do Bacharelado em Ciência da Computação

4.4 Matriz Curricular

A matriz curricular completa é apresentada na Figura 4.2. As linhas se referem aos oito semestres do curso (de S1 a S8), indicados pelos S# do lado esquerdo da figura. As colunas delimitam os sete núcleos de conhecimento citados anteriormente. Os semestres S3 a S8 também são aqueles nos quais os estudantes estarão envolvidos em atividades complementares e de extensão, conforme indicado na parte inferior da figura.

As caixas com bordas cheias representam as disciplinas obrigatórias, isto é, aquelas que todo estudante deve cursar. As caixas com bordas tracejadas são disciplinas optativas que podem ser escolhidas pelo estudante para configurar seu curso. Dentro de cada caixa representando cada disciplina observa-se uma numeração no formato #|#|#|#. Essa numeração indica a quantidade de horas total da disciplina, dividida em carga horária teórica, prática (em laboratório), de extensão ou estágio, nesta ordem. Por exemplo, a disciplina Construção de Algoritmos e Programação (CAP) possui carga horária de 120 horas, sendo 60 teóricas e 60 práticas; enquanto a disciplina Computação e Sociedade possui carga horária de 60 horas de extensão.

Nesta figura pode-se visualizar também o número de disciplinas por semestre e também a carga horária de cada semestre. Por exemplo, no primeiro semestre do curso (S1) há cinco disciplinas obrigatórias que precisam ser cursadas pelo estudante e isso contabiliza 390 horas. A carga horária de cada semestre pode ser vista embaixo da sigla S# que indica o número do semestre.

As disciplinas optativas profissionalizantes (caixas com bordas tracejadas) permitem que o estudante possa configurar seu curso da maneira como desejar a partir do quinto semestre. Por exemplo, no 5º semestre (S5), espera-se que o estudante curse 300 horas em disciplinas, como pode ser visto pelo número embaixo da sigla S#. Entretanto, nesse semestre há apenas duas disciplinas obrigatórias (Cálculo Numérico e Redes de Computadores) que somam 120 horas, restando 180 horas (equivalente a 3 disciplinas de 60 horas) para serem completadas com disciplinas livremente escolhidas pelo estudante para configurar seu semestre.

	Fundamentos de Matemática e Estatística	Fundamentos de Ciência da Computação	Algoritmos e Programação	Metodologias e Técnicas da Computação				Sistemas de Computação			Formação Multidisciplinar e Humanística	Atividades Complementares	Atividades de Extensão	Estágio ou TCC						
Sem. 1 390	Cálculo Diferencial e Integral 1 75 15	Geometria Analítica 45 15	Estatística Básica 60	Matemática Discreta 60	Introdução ao Pensamento Algorítmico 30	Construção de Algoritmos e Programação 60 60			Lógica Digital 60 30											
Sem. 2 390		Lógica Matemática 60			Algoritmos e Estruturas de Dados 1 60	Programação Orientada a Objetos 30 30			Arquitetura e Organização de Computadores 1 60 30											
Sem. 3 390	Cálculo Diferencial e Séries 45 15				Algoritmos e Estruturas de Dados 2 60	Organização e Recuperação da Informação 60			Sistemas Operacionais 60 30											
Sem. 4 360	Álgebra Linear 1 45 15				Projeto e Análise de Algoritmos 60	Programação Orientada a Objetos Avançada 30 30	Interação Humano-Computador 30 30	Engenharia de Software 1 60	Banco de Dados 30 30	Inteligência Artificial 30 30										
Sem. 5 300	Cálculo Numérico 45 15				Projeto e Análise de Algoritmos 2 60		Desenvolvimento de Software para Web 1 60	Engenharia de Software 2 60	Projeto e Implementação de Banco de Dados 30 30	Aprendizado de Máquina 1 60	Computação Gráfica 30 30	Processamento Digital de Imagens 30 30	Redes de Computadores 60	Sistemas Distribuídos 60	Processamento de Sinais Digitais 60 30					
Sem. 6 300	Otimização Matemática 60				Paradigmas de Linguagens de Programação 30 30		Desenvolvimento de Software para Web 2 60	Arquitetura de Software e Padrões 60	Banco de Dados para Ciência de Dados 30 30	Aprendizado de Máquina 2 60	Processamento e Visualização de Dados 30 30	Processamento Digital de Imagens 3D e Vídeos 30 30	Programação Paralela e Distribuída 30 30	Arquitetura e Organização de Computadores 2 30 30	Sistemas Digitais 30 30	Segurança Cibernética 30 30				
Sem. 7 240		Teoria da Computação 60					Desenvolvimento Móvel 60	DevOps 60	Processamento de Linguagem Natural 60	Visão Computacional 30 30			Projeto de Sistemas Computacionais Embaçados 60	Arquiteturas de Alto Desempenho 30 30	Processamento de Dados em Escala 30 30	Tecnologia de Comunicação 30 30				
Sem. 8 870		Construção de Compiladores 60												Atividades Complementares 210	Atividades Curriculares de Extensão 300	Estágio em Computação 360	Trabalho de Conclusão de Curso 360			

Figura 4.2: Matriz curricular completa

A Figura 4.3 apresenta uma visão diferente da matriz curricular apresentada na Figura 4.2, pois ela apresenta a visão do estudante, ou seja, as disciplinas obrigatórias e a quantidade de optativas de trilha (profissionalizantes) que devem ser cursadas. Nessa visão, fica claro que o estudante deve cursar 9 disciplinas optativas de trilha, escolhendo a partir das várias opções disponíveis mostradas na Figura 4.2.

Sem. 1	Cálculo Diferencial e Integral 1	Matemática Discreta	Introdução ao Pensamento Algorítmico	Construção de Algoritmos e Programação	Lógica Digital	
390	75 15	60	30	60 60	60 30	
Sem. 2	Geometria Analítica	Estatística Básica	Lógica Matemática	Algoritmos e Estruturas de Dados 1	Programação Orientada a Objetos	Arquitetura e Organização de Computadores 1
390	45 15	60	60	60	30 30	60 30
Sem. 3	Cálculo Diferencial e Séries	Algoritmos e Estruturas de Dados 2	Organização e Recuperação da Informação	Sistemas Operacionais	Metodologia Científica	Computação e Sociedade
390	45 15	60	60	60 30	60	60
Sem. 4	Álgebra Linear 1	Projeto e Análise de Algoritmos	Interação Humano-Computador	Engenharia de Software 1	Banco de Dados	Inteligência Artificial
360	45 15	60	30 30	60	30 30	30 30
Sem. 5	Cálculo Numérico	Disciplina Optativa	Disciplina Optativa	Disciplina Optativa	Redes de Computadores	
300	45 15				60	
Sem. 6	Disciplina Optativa	Teoria da Computação	Disciplina Optativa	Disciplina Optativa	Programação Paralela e Distribuída	
300		60			30 30	
Sem. 7	Construção de Compiladores	Disciplina Optativa	Disciplina Optativa	Disciplina Optativa		
240	60					
Sem. 8	Atividades Complementares	Atividades Curriculares de Extensão	Estágio em Computação	Trabalho de Conclusão de Curso		
870	210	300	360	360		

Figura 4.3: Matriz curricular completa na visão do estudante

Os Quadros a seguir trazem o detalhamento da matriz curricular indicando, para cada semestre do curso, as disciplinas, a sigla que as representa, o caráter (Obrigatório, Optativo),

os pré-requisitos, o departamento ofertante e a carga horária de cada uma dividida em horas teóricas (T), práticas (P), extensionistas (Ex) ou de Estágio (Es).

Quadro 4.2: Matriz Curricular do 1º Semestre (390 horas).

nº	Disciplina	Requisitos	Deptº. Ofertante	Caráter	Natureza (horas)				
					T	P	Ex	Es	Total
1	Cálculo Diferencial e Integral 1	—	DM	Obrigatório	75	15	—	—	90
2	Matemática Discreta	—	DC	Obrigatório	60	—	—	—	60
3	Introdução ao Pensamento Algorítmico	—	DC	Obrigatório	30	—	—	—	30
4	Construção de Algoritmos e Programação	—	DC	Obrigatório	60	60	—	—	120
5	Lógica Digital	—	DC	Obrigatório	60	30	—	—	90

Quadro 4.3: Matriz Curricular do 2º Semestre (390 horas).

nº	Disciplina	Requisitos	Deptº. Ofertante	Caráter	Natureza (horas)				
					T	P	Ex	Es	Total
1	Geometria Analítica	—	DM	Obrigatório	45	15	—	—	60
2	Estatística Básica	—	DEs	Obrigatório	60	—	—	—	60
3	Lógica Matemática	—	DC	Obrigatório	60	—	—	—	60
4	Algoritmos e Estruturas de Dados 1	Construção de Algoritmos e Programação	DC	Obrigatório	60	—	—	—	60
5	Programação Orientada a Objetos	Construção de Algoritmos e Programação	DC	Obrigatório	30	30	—	—	60
6	Arquitetura e Organização de Computadores 1	Lógica Digital	DC	Obrigatório	60	30	—	—	90

Quadro 4.4: Matriz Curricular do 3º Semestre (390 horas).

nº	Disciplina	Requisitos	Deptº. Ofertante	Caráter	Natureza (horas)				
					T	P	Ex	Es	Total
1	Cálculo Diferencial e Séries	Cálculo 1	DM	Obrigatório	45	15	—	—	60
2	Algoritmos e Estruturas de Dados 2	Algoritmos e Estruturas de Dados 1	DC	Obrigatório	60	—	—	—	60
3	Organização e Recuperação da Informação	Algoritmos e Estruturas de Dados 1	DC	Obrigatório	60	—	—	—	60
4	Sistemas Operacionais	Arquitetura e Organização de Computadores 1	DC	Obrigatório	60	30	—	—	90
5	Computação e Sociedade	—	DC	Obrigatório	—	—	60	—	60
6	Metodologia Científica	—	DC	Obrigatório	60	—	—	—	60

49

Quadro 4.5: Matriz Curricular do 4º Semestre (360 horas).

nº	Disciplina	Requisitos	Deptº. Ofertante	Caráter	Natureza (horas)				
					T	P	Ex	Es	Total
1	Álgebra Linear 1	Geometria Analítica	DM	Obrigatório	45	15	—	—	60
2	Projeto e Análise de Algoritmos	Algoritmos e Estruturas de Dados 2	DC	Obrigatório	60	—	—	—	60
3	Interação Humano-Computador	Construção de Algoritmos e Programação	DC	Obrigatório	30	30	—	—	60
4	Engenharia de Software 1	Programação Orientada a Objetos	DC	Obrigatório	60	—	—	—	60
5	Banco de Dados	Algoritmos e Estrutura de Dados 1	DC	Obrigatório	30	30	—	—	60
6	Inteligência Artificial	Algoritmos e Estruturas de Dados 1	DC	Obrigatório	30	30	—	—	60

Quadro 4.6: Matriz Curricular do 5º Semestre (300 horas).

nº	Disciplina	Requisitos	Deptº. Ofertante	Caráter	Natureza (horas)				
					T	P	Ex	Es	Total
1	Cálculo Numérico	Cálculo Diferencial e Integral 1, Geometria Analítica e Construção de Algoritmos e Programação	DM	Obrigatório	45	15	—	—	60
2	Redes de Computadores	Sistemas Operacionais	DC	Obrigatório	60	—	—	—	60
3	Disciplina Optativa	—	DC	Optativo	—	—	—	—	60
4	Disciplina Optativa	—	DC	Optativo	—	—	—	—	60
5	Disciplina Optativa	—	DC	Optativo	—	—	—	—	60

Quadro 4.7: Matriz Curricular do 6º Semestre (300 horas).

nº	Disciplina	Requisitos	Deptº. Ofertante	Caráter	Natureza (horas)				
					T	P	Ex	Es	Total
1	Programação Paralela e Distribuída	Sistemas Operacionais	DC	Obrigatório	30	30	—	—	60
2	Teoria da Computação	Matemática Discreta	DC	Obrigatório	60	—	—	—	60
3	Disciplina Optativa	—	DC	Optativo	—	—	—	—	60
4	Disciplina Optativa	—	DC	Optativo	—	—	—	—	60
5	Disciplina Optativa	—	DC	Optativo	—	—	—	—	60

Quadro 4.8: Matriz Curricular do 7º Semestre (240 horas).

nº	Disciplina	Requisitos	Deptº Ofertante	Caráter	Natureza (horas)				
					T	P	Ex	Es	Total
1	Construção de Compiladores	Teoria da Computação e Construção de Algoritmos e Programação	DC	Obrigatório	60	—	—	—	60
2	Disciplina Optativa	—	DC	Optativo	—	—	—	—	60
3	Disciplina Optativa	—	DC	Optativo	—	—	—	—	60
4	Disciplina Optativa	—	DC	Optativo	—	—	—	—	60

Quadro 4.9: Matriz Curricular do 8º Semestre (escolher entre Estágio ou TCC: 360 horas).

nº	Disciplina	Requisitos	Deptº Ofertante	Caráter	Natureza (horas)				
					T	P	Ex	Es	Total
1	Estágio em Computação	Ter sido aprovado em no mínimo 1830 horas	DC	Obrigatório	—	—	—	360	360
2	Trabalho de Conclusão de Curso	Ter sido aprovado em no mínimo 2310 horas	DC	Obrigatório	360	—	—	—	360

Quadro 4.10: Matriz Curricular – Consolidação da Formação (atividades realizadas durante todo o curso: 510 horas).

nº	Disciplina	Requisitos	Deptº Ofertante	Caráter	Natureza (horas)				
					T	P	Ex	Es	Total
1	Atividades Curriculares de Extensão	(depende da atividade)	(depende da atividade)	Obrigatório	—	—	300	—	300
2	Atividades Complementares	(depende da atividade)	(depende da atividade)	Obrigatório	210	—	—	—	210

Quadro 4.11: Matriz Curricular com as Optativas Recomendadas para o 5º Semestre (escolher três disciplinas: 180 h).

nº	Disciplina	Requisitos	Deptº Ofertante	Caráter	Natureza (horas)				Total
					T	P	Ex	Es	
1	Projeto e Análise de Algoritmos 2	Projeto e Análise de Algoritmos	DC	Optativo	60	–	–	–	60
2	Programação Orientada a Objetos Avançada	Algoritmos e Estruturas de Dados 1 e Programação Orientada a Objetos	DC	Optativo	30	30	–	–	60
3	Desenvolvimento de Software para Web 1	Banco de Dados	DC	Optativo	–	60	–	–	60
4	Engenharia de Software 2	Engenharia de Software 1	DC	Optativo	60	–	–	–	60
5	Projeto e Implementação de Banco de Dados	Banco de Dados	DC	Optativo	30	30	–	–	60
6	Aprendizado de Máquina 1	Inteligência Artificial e Estatística Básica	DC	Optativo	60	–	–	–	60
7	Computação Gráfica	Geometria Analítica e Algoritmos e Estruturas de Dados 1	DC	Optativo	30	30	–	–	60
8	Processamento Digital de Imagens	Construção de Algoritmos e Programação, Cálculo Diferencial e Integral 1 e Geometria Analítica	DC	Optativo	30	30	–	–	60
9	Sistemas Distribuídos	Sistemas Operacionais	DC	Optativo	60	–	–	–	60
10	Processamento de Sinais Digitais	Cálculo Diferencial e Integral 1, Geometria Analítica, Álgebra Linear e Construção de Algoritmos e Programação	DC	Optativo	60	30	–	–	90

Quadro 4.12: Matriz Curricular com as Optativas Recomendadas para o 6º Semestre (escolher três disciplinas: 180 h).

nº	Disciplina	Requisitos	Deptº. Ofertante	Caráter	Natureza (horas)					Total
					T	P	Ex	Es	Total	
53	1	Paradigmas de Linguagens de Programação	Projeto e Análise de Algoritmos	DC	Optativo	30	30	–	–	60
	2	Desenvolvimento de Software para Web 2	Desenvolvimento de Software para Web 1	DC	Optativo	–	60	–	–	60
	3	Arquitetura de Software e Padrões	Engenharia de Software 2	DC	Optativo	60	–	–	–	60
	4	Banco de Dados para Ciência de Dados	Projeto e Implementação de Banco de Dados	DC	Optativo	30	30	–	–	60
	5	Aprendizado de Máquina 2	Aprendizado de Máquina 1	DC	Optativo	60	–	–	–	60
	6	Processamento e Visualização de Dados	—	DC	Optativo	30	30	–	–	60
	7	Processamento Digital de Imagens 3D e Vídeos	Processamento Digital de Imagens	DC	Optativo	30	30	–	–	60
	8	Arquitetura e Organização de Computadores 2	Arquitetura e Organização de Computadores 1	DC	Optativo	30	30	–	–	60
	9	Sistemas Digitais	Lógica Digital	DC	Optativo	30	30	–	–	60
	10	Segurança Cibernética	Sistemas Operacionais e Arquitetura e Organização de Computadores 1	DC	Optativo	30	30	–	–	60
	11	Otimização Matemática	Cálculo 2	DC	Optativo	60	–	–	–	60

Quadro 4.13: Matriz Curricular com as Optativas Recomendadas para o 7º Semestre (escolher três disciplinas: 180 h).

nº	Disciplina	Requisitos	Deptº. Ofertante	Caráter	Natureza (horas)				Total
					T	P	Ex	Es	
1	Desenvolvimento Móvel	Desenvolvimento de Software para Web 2	DC	Optativo	–	60	–	–	60
2	DevOps	Engenharia de Software 2	DC	Optativo	–	60	–	–	60
3	Processamento de Linguagem Natural	Inteligência Artificial	DC	Optativo	60	–	–	–	60
4	Visão Computacional	Processamento de Sinais Digitais ou (Álgebra Linear e Processamento Digital de Imagens), Aprendizado de Máquina 2	DC	Optativo	30	30	–	–	60
5	Projeto de Sistemas Computacionais Embarcados	Arquitetura e Organização de Computadores 2 e Engenharia de Sistemas	DC	Optativo	–	60	–	–	60
6	Arquiteturas de Alto Desempenho	Arquitetura e Organização de Computadores 1	DC	Optativo	30	30	–	–	60
7	Processamento de Dados em Escala	Sistemas Operacionais	DC	Optativo	30	30	–	–	60
8	Tecnologia de Comunicação	Sistemas Operacionais	DC	Optativo	30	30	–	–	60

O Quadro 4.14 mostra o conjunto de disciplinas eletivas de formação multidisciplinar e humanística recomendadas para este curso, bem como o departamento ofertante, sua carga horária e o semestre no qual sua realização é indicada. Estas disciplinas, quando realizadas pelo estudante, contabilizam horas de Atividade Complementar (veja Apêndice IV) podendo contabilizar carga horária de extensão caso esteja previsto na disciplina.

Quadro 4.14: Eletivas de Formação Multidisciplinar e Humanística

Eletivas de Formação Multidisciplinar e Humanística	Departamento	Horas				Sem.
		T	P	Ex	Total	
Sociologia das relações raciais e estudos afro-brasileiros (1000955)	Sociologia	60	0	0	60	5-8
		60	0	0	60	5-8
		60	0	0	60	5-8
		60	0	0	60	5-8
		30	30	0	60	5-8
Gestão de projetos em unidades de informação (300470)	Ciência da Informação	30	30	0	60	5-8
		30	30	0	60	5-8
		60	0	0	60	5-8
		30	30	0	60	5-8
Introdução à língua brasileira de sinais LIBRAS (1002463)	Psicologia	60	0	0	60	5-8
Economia geral (164003)	Ciências Sociais	60	0	0	60	5-8
		60	0	0	60	5-8
		60	0	0	60	5-8
Estudos sociopolíticos dos algoritmos e da inteligência artificial (1002244)		60	0	0	60	5-8
		60	0	0	60	5-8
História social do Brasil (162116)		60	0	0	60	5-8
		60	0	0	60	5-8
		60	0	0	60	5-8
Filosofia da ciência (180025)	Filosofia	60	0	0	60	5-8
		60	0	0	60	5-8
Introdução à filosofia (180041)		30	30	0	60	5-8
		30	30	0	60	2-8
		30	30	0	60	2-8
Comunicação e expressão (62014)	Letras	45	15	0	60	5-8
Ingles Instrumental para Computação 1 (61085)		60	0	0	60	5-8
Ingles Instrumental para Computação 2 (61093)		0	0	60	60	2
Educação ambiental (550230)	Ciênc. Ambient.	45	15	0	60	5-8
Empreendedores em Informática (27090)	Computação	60	0	0	60	5-8
Ética em Computação (1002580)		0	0	60	60	2

Além das disciplinas previstas na matriz curricular, a seguir são apresentadas ACIEPEs geralmente cursadas pelos estudantes do BCC para complementar a formação com horas em atividades complementares extensionistas. O Quadro 4.15 traz algumas das ACIEPEs nas quais estudantes do BCC participaram em 2023. Essas são ACIEPEs recomendadas para os estudantes do BCC caso queiram utilizá-las como ACEs do tipo III para integralização da quantidade de horas em atividades complementares de extensão. A lista completa de ACIEPEs oferecidas na UFSCar pode ser obtida em <https://www.proex.ufscar.br/aciepes>.

Quadro 4.15: ACIEPEs frequentemente realizadas pelos estudantes do BCC e recomendadas para o curso (ACEs do tipo III)

Ativ. Curriculares de Integração entre Ensino, Pesquisa e Extensão		Horas			
ACIEPE	Departamento	T	P	Ex	Total
1	Empreendedorismo e Startups Mulheres na Ciência e Tecnologia: questões históricas e contemporâneas	Ciência da Informação (DCI)	0	0	60
2			0	0	60
3	Autocuidado com psicoterapia baseada em meditação	Medicina (DMed)	0	0	60
4	Acessibilidade e Inclusão de Pessoas com Deficiência no Trabalho	Engenharia de Produção (DEP)	0	0	60

4.5 Ementário

Esta seção apresenta, para cada semestre do curso, a descrição – objetivo e ementa – de cada disciplina. Algumas disciplinas possuem código porque já existem no sistema de gestão acadêmica da universidade (SIGA). As que não possuem código são disciplinas novas que ainda não estão cadastradas no SIGA.

4.5.1 Disciplinas do Primeiro Semestre

Título	Cálculo Diferencial e Integral 1
Objetivo Geral	A disciplina tem por objetivo levar os alunos a: interagir com fontes diretas e indiretas, selecionando e examinando criticamente tais fontes de modo a conduzir a uma prática de aprendizado continuado e autônomo em Cálculo; dominar conhecimentos e habilidades do Cálculo de uma variável relacionando-os com áreas correlatas como física, engenharia, estatística, química, e ciências da natureza, através da modelagem, resolução e análise de tais modelos; criar e demonstrar resultados simples em Cálculo e áreas correlatas; reconhecer a existência de características típicas de cálculo (funções de uma variável, limites, derivadas, integrais, etc) em problemas e as utilizar adequadamente.
Pré-requisitos	—
Disc. recomendadas	—
Carga horária total	90h
Natureza	▷ Teóricas: 75h ▷ Práticas: 15h
Caráter	Obrigatório

Ementa

Motivação do conceito de derivada e limite. Investigação de limites e suas aplicações para o estudo de continuidade de funções reais, para a caracterização de descontinuidades removíveis, e para a identificação de assíntotas verticais/horizontais. Caracterização geométrica e analítica da derivada como taxa de variação instantânea e como coeficiente angular da reta tangente, com interpretação em contextos físicos e geométricos. Desenvolvimento e prática das técnicas de derivação para funções algébricas, trigonométricas, exponenciais e logarítmicas, incluindo regra da cadeia, derivação implícita e derivadas de funções inversas. Aplicação da regra de L'Hôpital na resolução de limites indeterminados. Análise do comportamento de funções através do estudo de extremos locais, intervalos de crescimento e decrescimento, concavidade e pontos de inflexão, com aplicação ao esboço de gráficos. Modelagem de problemas de otimização contextualizados nas áreas de engenharia e ciências exatas, técnicas de solução e interpretação dos resultados. Contextualização de primitivas e de integrais indefinidas. Introdução ao conceito de integral definida através de somas de Riemann e sua interpretação geométrica como área sob uma curva. Estabelecimento de relações entre derivadas e integrais através do Teorema Fundamental do Cálculo. Prática das técnicas de substituição/mudança de variável, integração por partes e frações parciais. Aplicação de integrais, incluindo integrais impróprias, para o cálculo de áreas/volumes e para grandezas físicas diversas como distâncias, velocidades, trabalho e pressão.

Respons. pela oferta DM (1003500)

Bibliografia Básica

GUIDORIZZI, H. L. **Um curso de cálculo.** v. 1, 5. ed. Rio de Janeiro: Livros Técnicos e Científicos, 2001.

STEWART, J. **Cálculo.** v. 1, 5. ed. São Paulo: Pioneira Thomson Learning, 2006.

SWOKOWSKI, E. W. **Cálculo com geometria analítica.** v. 1, 2. ed. São Paulo: McGraw-Hill do Brasil, 1994.

Bibliografia Complementar

ANTON, H. **Cálculo.** v. 1, 10. ed, Porto Alegre, RS: Bookman, 2014.

ÁVILA, G. **Cálculo: funções de uma variável.** v. 1, 6. ed. Rio de Janeiro: Livros Técnicos e Científicos, 1994.

FLEMMING, D. M.; GONCALVES, M. B. **Cálculo A: funções, limite, derivação, integração.** 6. ed. São Paulo: Prentice Hall, 2006.

LEITHOLD, L. **O cálculo com geometria analítica.** v. 1, 3. ed. São Paulo: Harbra, 1991.

THOMAS, G. B. et al. **Cálculo.** v.1, 10. ed. Addison Wesley, 2002.

Título	Matemática Discreta
Objetivo Geral	Familiarizar os estudantes com a estrutura das demonstrações matemáticas, através da apresentação de diversos exemplos e exercícios; capacitar os estudantes a deduzir e utilizar fatos e noções elementares sobre números, conjuntos, relações, funções e grafos; estimular os estudantes a utilizar raciocínio indutivo em suas análises.
Pré-requisitos	—
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Teóricas: 60h
Caráter	Obrigatório
Ementa	Introdução à matemática discreta. Apresentação de estratégias de demonstração de teoremas com detalhamento de indução matemática. Introdução à teoria dos conjuntos, teoria dos números, somatórios e produtórios, com apresentação de propriedades matemáticas e demonstrações das mesmas. Apresentação de relações, relações de equivalência e relações de ordem. Noções de funções, funções injetoras, funções sobrejetoras e funções bijetoras. Introdução a grafos com apresentação de conceitos, como: conectividade e subgrafos, orientação e caminhos, graus e cortes, laços e arestas paralelas, coloração; além da introdução de categorias de grafos, como: árvores e circuitos, grafos bipartidos, eulerianos e hamiltonianos, grafos planares e duais. Apresentação de contagem (permutações, combinações e arranjos) e cardinalidade de conjuntos (finitos e infinitos). Problematização com exemplos práticos da computação.
Respons. pela oferta	DC (NOVA)
Bibliografia Básica	
GERSTING, J. L. Fundamentos matemáticos para a ciência da computação : um tratamento moderno de matemática discreta. 5. ed. Rio de Janeiro: LTC, c2004.	
SCHEINERMAN, E. R. Matemática discreta: uma introdução . São Paulo: Thomson Learning Edições, 2006.	
GOODAIRE, E. G. Goodaire; PARMENTER, Michael M. Discrete mathematics with graph theory . 3. ed. Upper Saddle River, NJ: Pearson Prentice Hall, c2006.	
Bibliografia Complementar	
ROSEN, K. H. Discrete mathematics and its applications . 7. ed. New York: McGraw Hill, 2013.	
GOMIDE, A.; STOLFI, J. Elementos de Matemática Discreta para Computação. 238 p. 2011.	
VELLEMAN D. How to Prove It, A Structured Approach . 2. ed. Cambridge, 2006.	
LEHMAN, E.; LEIGHTON, F. T.; MEYER, A. R. Mathematics for Computer Science . 2017.	
STEIN, C.; DRYSDALE, R. L.; BOGART, K. Matemática discreta para ciência da computação . São Paulo: Pearson Education do Brasil, 2013.	
FEOFILOFF, P.; KOHAYAKAWA, Y.; WAKABAYASHI, Y. Uma Introdução Sucinta à Teoria dos Grafos . 2011.	

Título	Introdução ao Pensamento Algorítmico
Objetivo Geral	Motivar e orientar os estudantes a desenvolver soluções sistemáticas para problemas diversos, contextualizados em situações cotidianas, de modo que estas possam ser implementadas a fim de usar o computador como ferramenta para obtenção de resultados. Desenvolver nos estudantes a habilidade de organizar e analisar dados de um problema, a fim de encontrar soluções utilizando técnicas de abstração, decomposição, reconhecimento de padrões e generalização, além da capacidade de analisar a eficiência de suas soluções.
Pré-requisitos	—
Disc. recomendadas	—
Carga horária total	30h
Natureza	▷ Teóricas: 30h
Caráter	Obrigatório
Ementa	Introdução ao pensamento algorítmico. Análise e especificação de problemas sob o aspecto de pensamento algorítmico. Técnicas de resolução de problemas: abstração, decomposição, reconhecimento de padrões e generalização. Representação e visualização de dados e soluções, com interpretação de resultados. Noções de paralelização. Noções de eficiência de um algoritmo. Introdução em alto nível de algoritmos de diversas áreas da ciência da computação: ordenação, busca, conectividade em grafos, caminhos mínimos, hashing, k-nn, criptografia.
Respons. pela oferta	DC (1001349)
Bibliografia Básica	<p>BHARGAVA, A. Y. Entendendo Algoritmos: Um guia ilustrado para programadores e outros curiosos. NOVATEC, 2017. 264 p. ISBN 978-85-752-2563-9.</p> <p>LOPES, A.; GARCIA, G. Introdução à programação: 500 algoritmos resolvidos. Rio de Janeiro: Elsevier, 2002. 469 p. ISBN 978-85-352-1019-4.</p> <p>SPRAUL, V. A. <i>Think like a programmer: an introduction to creative problem solving</i>. No Starch Press, 2012. 256 p. ISBN 978-1593274245</p> <p>SOUZA, M. A. F. Algoritmos e lógica de programação: um texto introdutório para engenharia. 2.ed. São Paulo: Cengage Learning, 2014. 234 p. ISBN 9788522111299.</p>
Bibliografia Complementar	<p>FORBELLONE, A. L. V.; EBERSPACHER, H. F. Lógica de programação: a construção de algoritmos e estruturas de dados. 3. ed. São Paulo: Pearson Prentice Hall, 2008. 218 p. ISBN 978-85-7605-024-7.</p> <p>HOLLOWAY, J. P. Introdução a programação para engenharia: resolvendo problemas com algoritmos. Rio de Janeiro: LTC, 2006. 339 p. ISBN 8521614535.</p> <p>SKIENA, S. S. <i>The algorithm design manual</i>. New York: Springer-Verlag, c1998. 486 p. ISBN 0-387-94860-0.</p> <p>ERWIG, Martin. <i>Once Upon an Algorithm: How Stories Explain Computing</i>. MIT Press, 2017. 336 p. ISBN 978-0262036634</p> <p>FILHO, W. F.; PICTET, R. <i>Computer Science Distilled: Learn the Art of Solving Computational Problems</i>. Code Energy, 2017. 183 p. ISBN 0997316004</p> <p>TILLMAN, F. A.; CASSONE, D. T. A <i>Professional's Guide to Problem Solving with Decision Science</i>. Pioneering Partnerships LLC, 2018. 298 p. ISBN 978-0999767115</p>

Título	Construção de Algoritmos e Programação
Objetivo Geral	Tornar os estudantes aptos a utilizar pensamento computacional e algorítmico para proposição de soluções de problemas. Capacitar os estudantes a mapear tais soluções em programas usando linguagem de programação.
Pré-requisitos	—
Disc. recomendadas	
Carga horária total	120h
Natureza	▷ Teóricas: 60h ▷ Práticas: 60h
Caráter	Obrigatório
Ementa	Noções gerais da computação: organização de computadores, programas, linguagens e aplicações. Detalhamento de algoritmos estruturados e programação: tipos básicos de dados. Representação e manipulação de dados. Estruturas de controle de fluxo (condicionais e repetições). Modularização (sub-rotinas, passagem de parâmetros e escopo). Documentação. Estruturação básica de dados: variáveis compostas heterogêneas (registros) e homogêneas (vetores e matrizes). Operações em arquivos e sua manipulação. Alocação dinâmica de memória e ponteiros. Noções de recursão.
Respons. pela oferta	DC (NOVA)
Bibliografia Básica	
CIFERRI, R.R. Programação de Computadores . Edufscar, 2009.	
MEDINA, M.; FERTIG, C. Algoritmos e Programação: Teoria e Prática , Novatec, 2005.	
SENNE, E. Primeiro Curso de Programação em C . Visual Books, 2003.	
TREMBLAY, J.P.; BUNT, R.B. Ciência dos Computadores : McGraw-Hill, 1981.	
KERNIGHAN, B.W.; RITCHIE, D.M. The C Programming Language . 2. ed. 1988.	
Bibliografia Complementar	
HARBISON, S.P.; STEELE., G.L. C: a reference manual . 2002.	
KOCHAN; S.G. Programming in C: A complete introduction to the C programming language . 2004.	
KING, K.N. C Programming: A Modern Approach . Norton & Company, 1996.	

Título	Lógica Digital
Objetivo Geral	A disciplina tem por objetivo promover nos alunos uma capacitação inicial para atuar no projeto e análise de circuitos digitais combinatórios e sequenciais, produzindo implementações usando circuitos integrados e linguagens de descrição de hardware, adquirindo ainda capacidade de aprender de forma autônoma conceitos teóricos e práticas associados à lógica digital.
Pré-requisitos	—
Disc. recomendadas	—
Carga horária total	90h
Natureza	▷ Teóricas: 60h ▷ Práticas: 30h
Caráter	Obrigatório
Ementa	A disciplina de Lógica Digital prevê a aquisição de conhecimentos, adquiridos a partir de diferentes fontes de estudo e prática, visando o projeto e análise de circuitos digitais combinatórios e sequenciais. Prevê o estudo e prática de conceitos fundamentais de eletrônica digital, como representação digital da informação; álgebra booleana; tabelas verdade e portas lógicas; expressões lógicas e formas canônicas; estratégias de minimização de circuitos. Prevê também estudo e práticas de laboratório envolvendo elementos de memória, máquinas de estado (Mealy e Moore) e circuitos funcionais típicos (combinacionais e sequenciais).
Respons. pela oferta	DC (1003451)
Bibliografia Básica	<p>TOCCI, R. J.; WIDMER, N. S. MOSS, G. L. Sistemas digitais: princípios e aplicações. 11. ed. São Paulo: Pearson Prentice Hall, 2011. p. 817. ISBN 9788576059226.</p> <p>WAKERLY, John F. Digital design: principles and practices. 4. ed. Upper Saddle River: Pearson Prentice Hall, 2006. 895 p. ISBN 0-13-186389-4.</p> <p>FLOYD, T. L. Sistemas digitais: fundamentos e aplicações. 9. ed. Porto Alegre, RS: Artmed, 2007. xiii, p.888. ISBN 9788560031931.</p>
Bibliografia Complementar	<p>BROWN, S.; VRANESIC, Z.; BROWN, S. Fundamentals of Digital Logic with Verilog Design. 2. ed. New York: McGraw-Hill Companies, 2007.</p> <p>PEDRONI, V. A. Eletrônica digital moderna e VHDL: princípios digitais, eletrônica digital, projeto digital, microeletrônica e VHDL. Rio de Janeiro: Elsevier, 2010. 619 p. ISBN 9788535234657.</p> <p>ERCEGOVAC, M. D.; LANG, T. Digital Arithmetic. San Francisco: Morgan Kaufmann, 2004. 709 p. ISBN 1-55860-798-6.</p> <p>NELSON, V. P.; NAGLE, H. T.; CARROLL, B. D.; IRWIN, D. Digital Logic Circuit Analysis and Design. 1. ed. Upper Saddle River: Prentice Hall, 1995.</p> <p>BALABANIAN, N.; CARLSON, B. Digital Logic Design Principles. 1. ed. New York: Wiley, 2000.</p> <p>CARPINELLI, John D. An Animated Introduction to Digital Logic Design; New Jersey Institute of Technology Library, 2023. Versão aberta (licença Creative Commons) disponível em: https://digitalcommons.njit.edu/oat/1/. Acesso em: 31/10/2024</p>

4.5.2 Disciplinas do Segundo Semestre

Título	Geometria Analítica
Objetivo Geral	A disciplina tem por objetivo levar os alunos a: interagir com fontes diretas e indiretas, selecionando e examinando criticamente tais fontes de modo a conduzir a uma prática de aprendizado continuado e autônomo em Geometria Analítica; dominar conhecimentos e habilidades de Geometria Analítica relacionando esses conhecimentos e habilidades com áreas correlatas como física, engenharia e estatística, através da modelagem, resolução e análise de tais modelos; criar e demonstrar resultados simples em Álgebra Linear, Geometria e áreas correlatas usando a correspondência entre técnicas e conceitos destas duas áreas da Matemática; reconhecer a existência de características típicas de Álgebra Linear (combinação linear, coordenadas em uma base, etc) e Geometria (relações entre pontos, retas e planos, congruências, ordenação no espaço, etc) em problemas e as utilizar adequadamente.
Pré-requisitos	—
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Teóricas: 45h ▷ Práticas: 15h
Caráter	Obrigatório
Ementa	Estudo do posto de matrizes e sua relação com sistemas lineares. Desenvolvimento das técnicas de resolução de sistemas lineares através dos métodos de Eliminação Gaussiana e forma de Gauss-Jordan. Caracterização de vetores no espaço tridimensional e dos conceitos de dependência e independência linear, bases e sistemas de coordenadas. Desenvolvimento das operações com vetores: produto escalar, produto vetorial e produto misto, com aplicações em problemas geométricos. Representações de retas e planos na forma paramétrica e como solução de sistema linear, investigação da posição relativa de retas e planos. Estudo de projeções ortogonais, distâncias e ângulos entre objetos geométricos. Caracterização e análise das cônicas (elipse, hipérbole e parábola) e a identificação de pontos e elementos especiais. Estabelecimento das relações entre as propriedades focais das cônicas e suas aplicações físicas. Investigação de cônicas através de mudanças de coordenadas e rotações. Introdução ao estudo de quâdricas, sua classificação, visualização e análise através de seções planas.
Respons. pela oferta	DM (1003457)
Bibliografia Básica	<p>BOULOS, P. e CAMARGO, I. Geometria analítica, um tratamento vetorial. 3. ed. Pearson Editora, 2005.</p> <p>CAROLI, A., CALLIOLI, C. A., FEITOSA, M. O. Matrizes, vetores e geometria analítica. Editora Nobel, São Paulo, 1987.</p> <p>STEINBRUCH, A., WINTERLE, P. Geometria analítica. 2. ed. Pearson Editora, São Paulo, 2006.</p>

Bibliografia Complementar

- BALDIN, Y. Y. e FURUYA, Y. K. S. **Geometria analítica para todos e atividades com Octave e GeoGebra.** São Carlos: EDUFSCa, 2011.
- CALLIOLI, C. A., DOMINGUES, H. H., COSTA, R. **Álgebra linear e aplicações.** 6 ed. São Paulo: Atual, 2007.
- LIMA, E. L. **Geometria analítica e álgebra Linear.** IMPA, 2001.
- LIPSCHUTZ, S. **Álgebra linear: teoria e problemas.** McGraw-Hill, 1994.
- WINTERLE, P. **Vetores e geometria analítica.** Makron Books, 2000.
-

Título	Estatística Básica
Objetivo Geral	
Os estudantes serão capazes de entender a aplicação dos conceitos e métodos estatísticos, visando a análise de dados provenientes de diferentes fontes, interpretando criticamente os resultados obtidos de uma análise estatística realizada de forma adequada, relatando-os e utilizando-os para a tomada de decisão.	
Pré-requisitos	—
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Teóricas: 60h
Caráter	Obrigatório
Ementa	Estudo do método estatístico, compreendendo fenômenos aleatórios e o pensamento estatístico. Estudo de ferramentas estatísticas para a descrição e resumo de dados, utilizando tabelas, gráficos e medidas descritivas. Estudo de conceitos envolvendo probabilidade e probabilidade condicional para o cálculo de probabilidades. Estudo de modelos probabilísticos e suas propriedades. Compreensão dos princípios de inferência estatística entendendo como relacionar os modelos probabilísticos com os problemas de inferência estatística. Aprender a utilizar e interpretar estimativas pontual e intervalar para a média populacional. Aprender a estruturar problemas de tomada de decisão como um teste de hipótese, utilizando e interpretando testes de hipóteses para a média populacional e para a variância populacional. Estudo de modelo de regressão linear simples, sabendo utilizar e interpretar os resultados para predição, verificando a adequabilidade e violação de suposições a partir da análise de seus resíduos.
Respons. pela oferta	DEs (1003522)
Bibliografia Básica	
MONTGOMERY, D.C., RUNGER, G.C. Estatística Aplicada e Probabilidade para Engenheiros . LTC Editora, 5. ed. Rio Janeiro, 2013.	
MORETTIN, P. A.; BUSSAB, W. O. Estatística Básica . 7. ed. São Paulo, Editora Saraiva, 2011.	
TRIOLA, M. F. Introdução à estatística . 7. ed. Rio de Janeiro, Editora LTC, 1999.	
WALPOLE, R. E.; MYERS, R. H.; MYERS, S. L.; YE, K. Probabilidade e estatística para engenharia e ciências . 8. ed. Pearson Prentice Hall, São Paulo, 2009.	
Bibliografia Complementar	

Título	Lógica Matemática
Objetivo Geral	Desenvolver nos estudantes a capacidade de raciocínio lógico e abstrato no intuito de capacitar o estudante a propor algoritmos rápidos e eficientes; ensinar aos estudantes os fundamentos sobre sistemas dedutivos e formalismos da lógica clássica; tornar os estudantes aptos a conhecer os conceitos da Lógica Proposicional e da Lógica de Primeira Ordem e suas aplicações na computação.
Pré-requisitos	—
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Teóricas: 60h
Caráter	Obrigatório
Ementa	Apresentação da Lógica proposicional: proposições atômicas, conectivos, proposições compostas, fórmulas bem formadas, linguagem proposicional, semântica (interpretações e modelos), consequência lógica, equivalência lógica, dedução, formas normais, notação clausal, cláusulas de Horn, regras de inferência, argumentos, o princípio da resolução. Apresentação da Lógica de primeira ordem (lógica de predicados): alfabetos, termos, fórmulas bem formadas, linguagem de primeira ordem, escopo de quantificadores, variáveis livres e ligadas, semânticas (modelos), consequência lógica, equivalência lógica, dedução, skolemização, formas normais, quantificação universal, notação clausal, cláusulas de Horn, substituição e unificação, unificadores mais gerais, o princípio de resolução.
Respons. pela oferta	DC (1001529)
Bibliografia Básica	
NICOLETTI, M.C. A Cartilha da Lógica. Série de Apontamentos, 2. ed. São Carlos: EdUFSCar, 2009. 233 p.	
SILVA, F. S. C.; FINGER, M., MELO, A.C.V. Lógica para computação. Thomson, 2010.	
SOUZA, J. N. Lógica para ciência da computação: uma introdução concisa. Elsevier, 2008.	
Bibliografia Complementar	
LEVADA, A. L. M. Fundamentos de lógica matemática. Coleção UAB-UFSCar, Sistemas de Informação. 2010, 170 pgs.	
GERSTING, J. L. Fundamentos Matemáticos para a Ciência da Computação: um tratamento moderno de matemática discreta. 5. ed. Rio de Janeiro: LTC, 2004.	

Título	Algoritmos e Estruturas de Dados 1
Objetivo Geral	A disciplina tem por objetivo promover nos estudantes versatilidade na construção de soluções algorítmicas para problemas computacionais fundamentais, bem como capacidade crítica para analisar pontos fortes e fracos destas soluções. Com isso, deve tornar os estudantes aptos a utilizar técnicas básicas de programação em seus projetos; capacitar os estudantes a reconhecer, implementar e modificar algoritmos e estruturas de dados básicos; familiarizar os estudantes com noções de projeto e análise de algoritmos, através do estudo de uma linguagem algorítmica, exemplos e exercícios práticos; estimular os estudantes a avaliar quais técnicas de programação, algoritmos e estruturas de dados se adéquam melhor a cada situação, problema ou aplicação.
Pré-requisitos	Construção de Algoritmos e Programação
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Teóricas: 60h
Caráter	Obrigatório
Ementa	Aprofundamento sobre recursão e o conhecimento sobre como aplicá-la em algoritmos. Noções sobre análise de correção (invariantes) e eficiência (complexidade) de algoritmos. Conhecimento e aplicação de algoritmos de busca linear e binária. Noções sobre programação por retrocesso (<i>backtracking</i>) e algoritmos de enumeração. Conhecimento acerca do conceito de tipos abstratos de dados. Conhecimento, utilização e manipulação de listas (alocação estática e dinâmica, circulares, duplamente ligadas e com nó cabeça). Conhecimento, utilização e manipulação de matrizes e listas ortogonais, pilhas e filas (implementação sequencial e ligada). Conhecimento, utilização e manipulação de árvores binárias e de árvores de busca auto-balanceadas do tipo AVL. Conhecimento, utilização e manipulação de filas de prioridade (<i>heap</i>). Apresentação de exemplos e exercícios práticos, os quais podem envolver estruturas de dados compostas (como vetores e listas ligadas) e diferentes abordagens algorítmicas (gulosa, divisão e conquista, programação dinâmica, <i>backtracking</i> , busca em largura, etc).
Respons. pela oferta	DC (NOVA)
Bibliografia Básica	
FEOFILOFF. P. Algoritmos em Linguagem C . Elsevier, 2009.	
TENENBAUM, A. M., LANGSAM, Y., AUGENSTEIN, M. J. Estruturas de dados usando C . São Paulo: Pearson Makron Books, 2009.	
FERRARI, R., RIBEIRO, M. X., DIAS, R. L., FALVO, M. Estruturas de Dados com Jogos . Rio de Janeiro: Elsevier, 2014.	
SEDGEWICK, R. <i>Algorithms in C++, Parts 1-4: fundamentals, data structures, sorting, searching</i> . 3. ed. Boston: Addison-Wesley, 1998.	
SEDGEWICK, R. <i>Algorithms in C++, Part 5: graph algorithms</i> . 3. ed. Boston: Addison-Wesley, 2001.	
SEDGEWICK, R. <i>Algorithms in Java : parts 1-4 : fundamentals, data structures, sorting, searching</i> . 3rd. ed., Boston: Addison - Wesley, 2003.	

Bibliografia Complementar

- ZIVIANI, N. **Projetos de algoritmos: com implementações em Pascal e C.** 3. ed. rev. e ampl. São Paulo: Cengage Learning, 2012.
- ZIVIANI, N. **Projeto de algoritmos: com implementações em Java e C++.** São Paulo: Thomson Learning, 2007.
- CORMEN, T. H., Leiserson, C. E., Rivest, R. L., Stein, C. *Introduction to Algorithms.* 3a edição, The MIT Press, 2009
- CORMEN, T. H. et al. **Algoritmos: teoria e prática.** Rio de Janeiro: Elsevier, 2012.
- MENEZES, N. N. C.. **Introdução à programação com Python: algoritmos e lógica de programação para iniciantes.** 2. ed. São Paulo: Novatec, 201.
-

Título	Programação Orientada a Objetos
Objetivo Geral	Capacitar os estudantes nos conceitos básicos de programação orientada a objetos e suas características principais. Capacitar os estudantes na construção de programas utilizando uma linguagem baseada no paradigma de orientação a objetos.
Pré-requisitos	Construção de Algoritmos e Programação
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Teóricas: 30h ▷ Práticas: 30h
Caráter	Obrigatório
Ementa	Histórico do paradigma orientado a objetos e comparação com o paradigma estruturado. Conceitos teóricos e práticos de orientação a objetos: abstração, classes, objetos, atributos e métodos, encapsulamento/visibilidade, herança, composição/agregação, sobrecarga, polimorfismo de inclusão e classes abstratas e polimorfismo paramétrico. Modularização. Alocação dinâmica de objetos. Tratamento de exceções.
Respons. pela oferta	DC (1001507)
Bibliografia Básica	
	DEITEL, H.M. & DEITEL, P. J. C++ Como Programar . 5. ed. Pearson Prentice Hall, 2006.
	PIZZOLATO, E. B. Introdução à programação orientada a objetos com C++ e Java . EdUFSCar, 2010.
	ECKEL, B. Thinking in C++ . 2. ed. Upper Saddle River: Prentice Hall, 2000.
Bibliografia Complementar	
	SILVA FILHO, A. M. Introdução à programação orientada a objetos com C++ . Elsevier, 2010.
	DEITEL, P. J.; DEITEL, H. M. C++ for programmers . Upper Saddle River, NJ: Prentice Hall, 2009. 1000 p. (Deitel Developer Series). ISBN 10-0-13-700130-9.
	SCHILD, H. C++: the complete reference . 4. ed. New York: McGraw Hill, 2003. 1023 p. ISBN 0-07-222680-3.

Título	Arquitetura e Organização de Computadores 1
Objetivo Geral	Ao final da disciplina o estudante deve ser capaz de entender os princípios da arquitetura e organização básica de computadores e a relação entre linguagens de alto nível e linguagens de máquina, bem como de criar um computador usando técnicas de implementação de unidades funcionais e analisar seu desempenho.
Pré-requisitos	Lógica Digital
Disc. recomendadas	—
Carga horária total	90h
Natureza	▷ Teóricas: 60h ▷ Práticas: 30h
Caráter	Obrigatório
Ementa	Conceitos fundamentais de Arquitetura de Computadores. Linguagem de máquina. Aritmética computacional. Organização do computador: monociclo, multiciclo e pipeline. Desempenho de computadores. Hierarquia de memória. Entrada/Saída: barramentos e dispositivos externos. Implementação de um processador completo usando linguagem de descrição de hardware.
Respons. pela oferta	DC (NOVA)
Bibliografia Básica	<p>PATTERSON, D. A.; HENNESSY, J. L. Organização e projeto de computadores: a interface hardware/software. 3. ed. Rio de Janeiro: Elsevier, 2005. 484 p. ISBN 8535215212.</p> <p>HARRIS, D. M.; HARRIS, S. L. Digital design and computer architecture. San Frasciso: Elsevier, 2007. 569 p. ISBN 978-0-12-370497-9.</p> <p>STALLINGS, W. Arquitetura e organização de computadores. 8. ed. São Paulo: Pearson, 2012. 624 p. ISBN 978-85-7605-564-8.</p> <p>SAITO, J. H. Introdução à arquitetura e à organização de computadores: síntese do processador MIPS. São Carlos, SP: EdUFSCar, 2010. 189 p. (Coleção UAB-UFSCar. Sistemas de Informação). ISBN 978-85-7600-207-9.</p>
Bibliografia Complementar	<p>HENNESSY, J. L.; PATTERSON, D. A. Arquitetura de computadores: uma abordagem quantitativa. 3. ed. Rio de Janeiro: Campus, 2003. 827 p. ISBN 85-352-1110-1.</p> <p>STALLINGS, W. Arquitetura e organização de computadores: projeto para o desempenho. 5. ed. São Paulo: Prentice Hall, 2002. 786 p. ISBN 85-87918-53-2.</p>

4.5.3 Disciplinas do Terceiro Semestre

Título	Cálculo Diferencial e Séries
Objetivo Geral	A disciplina tem por objetivo levar os alunos a: interagir com fontes diretas e indiretas, selecionando e examinando criticamente tais fontes de modo a conduzir a uma prática de aprendizado continuado e autônomo em Cálculo e Séries; dominar conhecimentos e habilidades de Cálculo diferencial de várias variáveis, e de séries numéricas, relacionando esses conhecimentos e habilidades com áreas correlatas como física e engenharia, através da modelagem, resolução e análise de tais modelos; criar e demonstrar resultados simples em Cálculo e Séries e áreas correlatas sob o ponto de vista de várias variáveis; reconhecer a existência de características típicas de Cálculo (funções de várias variáveis, gradientes, aproximação de Taylor, séries, etc) em problemas e as utilizar adequadamente.
Pré-requisitos	Cálculo 1
Disc. recomendadas	—
Carga horária total	60h
Natureza	► Teóricas: 45h ► Práticas: 15h
Caráter	Obrigatório
Ementa	Estudo das funções de várias variáveis reais e suas propriedades de continuidade e diferenciabilidade, assim como técnicas de visualização. Desenvolvimento dos conceitos de derivadas parciais, do vetor gradiente e derivadas direcionais, e suas interpretações geométricas. Introdução às técnicas de derivação implícita na análise de relações entre variáveis e suas interpretações geométricas e físicas. Investigação sobre a existência de funções implícitas e inversas e suas interpretações. Desenvolvimento da fórmula de Taylor para a aproximação de funções de várias variáveis seu uso na caracterização de pontos críticos como selas, máximos e mínimos locais; aplicações de Taylor em engenharia e física. Estudo da teoria de séries numéricas e de critérios de convergência como os testes de comparação e da integral. Investigação de séries de funções, como as séries de potências e as trigonométricas, com aplicações na aproximação de funções.
Respons. pela oferta	DM (1003663)
Bibliografia Básica	<p>GUIDORIZZI, H. L. Um curso de cálculo. v. 2, Livros Técnicos e Científicos, Rio de Janeiro, 2004</p> <p>PINTO, D.; MORGADO, M. C. F. Cálculo diferencial e integral de funções de várias variáveis. UFRJ/SR-1, 1997.</p> <p>THOMAS, G. B. et al. Cálculo, v.2, 10. ed., Addison Wesley, 2003.</p>
Bibliografia Complementar	<p>ÁVILA, G. S.S. Cálculo: funções de várias variáveis. v.3., 3. ed, Rio de Janeiro: Livros Técnicos e Científicos, 258 p., 1981.</p> <p>LIMA, E. L. Curso de análise. v.2, Projeto Euclides. Rio de Janeiro, IMPA, 1989.</p> <p>RUDIN, W. Principles of mathematical analysis. 3. ed. McGraw-Hill, 1976.</p> <p>STEWART, J. Cálculo. v. 2, 4. ed. Pioneira/Thomson Learning, São Paulo, 2001.</p> <p>SWOKOWSKI, E. W. Cálculo com geometria analítica, v. 2, 2. ed. Markron Books, 1991.</p>

Título	Algoritmos e Estruturas de Dados 2
Objetivo Geral	A disciplina tem por objetivo promover nos estudantes versatilidade na construção de soluções algorítmicas para problemas computacionais intermediários, bem como capacidade crítica para analisar pontos fortes e fracos destas soluções. Com isso, deve tornar os estudantes aptos a utilizar diversas técnicas de programação em seus projetos; capacitar os estudantes a reconhecer, implementar e modificar algoritmos e estruturas de dados amplamente utilizados; familiarizar os estudantes com o projeto e a análise de algoritmos, através do estudo de uma linguagem algorítmica, exemplos e exercícios práticos; estimular os estudantes a avaliar quais técnicas de programação, algoritmos e estruturas de dados se adéquam melhor a cada situação, problema ou aplicação.
Pré-requisitos	Algoritmos e Estruturas de Dados 1
Disc. recomendadas	—
Carga horária total	60h
Natureza	► Teóricas: 60h
Caráter	Obrigatório
Ementa	Aprofundamento das noções de análise de correção (invariantes e indução matemática) e eficiência (complexidade de tempo e espaço) de algoritmos, incluindo a notação O. Detalhamento de algoritmos de ordenação elementares, como <i>insertion sort</i> , <i>selection sort</i> ou <i>bubble sort</i> . Detalhamento dos algoritmos de ordenação não-elementares <i>heap sort</i> , <i>merge sort</i> e <i>quick sort</i> aleatorizado. Limitante inferior $\Omega(n \log n)$ para ordenação por comparação. Apresentação dos algoritmos de ordenação não baseados em comparação <i>counting sort</i> e <i>radix sort</i> . Introdução de tabelas de símbolos com detalhamento de sua implementação usando tabelas de espalhamento (<i>hash tables</i>). Detalhamento de outras implementações de tabelas de símbolos como árvores rubro-negras, <i>skip lists</i> , ou árvore digitais (básicas, Tries e PATRICIA Tries). Introdução a grafos com diferentes tipos (simples, dirigido e ponderado) e representações (matrizes e listas de adjacência). Detalhamento dos algoritmos de busca em largura e em profundidade em grafos (com aplicações como conectividade, caminhos mínimos não ponderados, ordenação topológica, detecção de pontes, componentes fortemente conexos). Detalhamento do algoritmo de Dijkstra (com e sem <i>heap</i>) para caminhos mínimos em grafos sem custos negativos. Apresentação de exemplos e exercícios práticos, os quais podem envolver estruturas de dados compostas (como <i>heaps</i> ou tabelas <i>hash</i>) e diferentes abordagens algorítmicas (gulosa, divisão e conquista, programação dinâmica, aleatorização, etc).
Respons. pela oferta	DC (NOVA)
Bibliografia Básica	<p>SEGEWICK, R. <i>Algorithms in C++, Part 5: graph algorithms.</i> 3. ed., Boston: Addison-Wesley, 2001.</p> <p>ZIVIANI, N. Projeto de algoritmos: com implementações em Java e C++. 2. ed. São Paulo: Cengage Learning, 2011.</p> <p>FEOFILOFF, P. Algoritmos em Linguagem C. Elsevier, 2009.</p> <p>CORMEN, T.H. ; LEISERSON, C.E. ; RIVEST, R.L.; STEIN, C. Introduction to Algorithms. 3. ed. McGraw-Hill, 2009.</p>

Bibliografia Complementar

- SEGEWICK, R. **Algorithms in C++, Parts 1-4: fundamentals, data structures, sorting, searching.** 3. ed. Boston: Addison - Wesley, 1998.
- BERMAN, A. M. **Data structures via C++: objects by evolution.** New York: Oxford University Press, 1997.
- LANGSAM, Y. ; AUGENSTEIN, M. ; TENENBAUM, A. M. **Data structures using C and C++.** 2. ed. Upper Sadle River: Prentice Hall, 1996.
- ZIVIANI, N. **Projetos de algoritmos:** com implementações em Pascal e C. 3. ed. rev. e ampl. São Paulo: Cengage Learning, 2012.
- DROZDEK, A. **Estruturas de dados e algoritmos em C++.** São Paulo: Cengage Learning, 2010.
-

Título	Organização e Recuperação da Informação
Objetivo Geral	Tornar os estudantes aptos a solucionar problemas que envolvem a organização e recuperação de informações armazenadas em arquivos. Capacitar os estudantes a implementar estruturas de dados adequadas à organização e busca de informação em meios externos. Familiarizar os estudantes com o projeto e a análise de algoritmos para lidar com informações em disco, através de exemplos e exercícios práticos. Estimular os estudantes a avaliar quais técnicas de programação, algoritmos e estruturas de dados se adequam melhor a cada situação, problema ou aplicação.
Pré-requisitos	Algoritmos e Estruturas de Dados 1
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Teóricas: 60h
Caráter	Obrigatório
Ementa	Apresentação dos conceitos de representação, organização, armazenamento e recuperação de dados em memória secundária. Noções sobre a estrutura física de dispositivos de armazenamento secundário (discos magnéticos, fitas magnéticas, discos de estado sólido e novas tecnologias). Apresentação do conceito de organização de arquivos: arquivos dos tipos binários e texto, campos, registros e reaproveitamento de espaço na remoção lógica de registros. Apresentação de conceitos e implementação de índices: índice linear, índice multinível, índices primário e secundário, estruturas de árvores de múltiplos caminhos (árvores B, B+, B* e B virtual com buffer-pool). Noções sobre índices para dados não convencionais (árvores métricas, quadtrees e índices bitmap). Apresentação de algoritmos para o processamento consequencial de listas em memória secundária e ordenação externa. Apresentação do conceito e implementação para hashing externo: funções e espalhamento, baldes, hash dinâmico, uso de hash como mecanismo de indexação. Apresentação de conceitos de compressão de dados sem perda de informação (Huffman, LZW ou similares). Apresentação da organização de memória interna: métodos sequenciais e não sequenciais; coleta de lixo.
Respons. pela oferta	DC (1001487)
Bibliografia Básica	<p>FOLK, M. J.; ZOELLICK, B. <i>File Structures</i>. 2. ed. Reading: Addison-Wesley, 1992.</p> <p>ZIVIANI, N. Projetos de algoritmos: com implementações em Pascal e C. 3. ed. rev. e ampl. São Paulo: Cengage Learning, 2012.</p> <p>DROZDEK, A. Estruturas de dados e algoritmos em C++. São Paulo: Cengage Learning, 2010.</p> <p>CORMEN, T. H.; LEISERSON, C. E.; RIVEST, R. L.; STEIN, C. <i>Introduction to Algorithms</i>. 3. ed. Boston: McGraw-Hill, 2009.</p>
Bibliografia Complementar	<p>LANGSAM, Y.; AUGENSTEIN, M. J.; TENENBAUM, A. M. <i>Data structures using C and C++</i>. 2. ed. Upper Saddle River: Prentice Hall, 1996.</p> <p>TENENBAUM, A. M.; LANGSAM, Y.; AUGENSTEIN, M. J. Estruturas de dados usando C. São Paulo: Pearson Makron Books, 2009.</p> <p>SZWARCFITER, J. L.; MARKENZON, L. Estruturas de Dados e seus Algoritmos. 3. ed. Rio de Janeiro: LTC, 2010.</p> <p>ZIVIANI, N. Projeto de algoritmos: com implementações em Java e C++. 2. ed. São Paulo: Cengage Learning, 2011.</p>

Título	Sistemas Operacionais
Objetivo Geral	Familiarizar os estudantes com Sistemas Operacionais, apresentando seus objetivos, suas funcionalidades e aspectos de suas organizações internas. Familiarizar os estudantes com as políticas para o gerenciamento de processos e recursos. Familiarizar os estudantes com as funcionalidades providas pelos Sistemas Operacionais como gerenciadores de recursos. Tornar o estudante ciente dos algoritmos e das abstrações utilizadas em projetos de sistemas operacionais para o gerenciamento de atividades a executar (processos e threads) e para o armazenamento de dados (arquivos). Habilitar o estudante a identificar os requisitos existentes para diferentes tipos de sistemas computacionais e suas implicações no projeto do sistema operacional (sistemas de tempo-real, servidores, dispositivos com capacidades de software e hardware limitadas). Tornar os estudantes aptos a criar programas que usem eficientemente os recursos e serviços providos por sistemas operacionais. Tornar os estudantes aptos a entender e atuar no projeto e no desenvolvimento de sistemas operacionais.
Pré-requisitos	Arquitetura e Organização de Computadores 1
Disc. recomendadas	—
Carga horária total	90h
Natureza	► Teóricas: 60h ► Práticas: 30h
Caráter	Obrigatório
Ementa	Introdução. Interface do SO. Processos, threads e gerenciamento do processador. Gerenciamento de memória. Comunicação e sincronização de processos e threads. Gerenciamento de armazenamento. Estudo de caso com sistemas operacionais.
Respons. pela oferta	DC (1001535)
Bibliografia Básica	<p>TANENBAUM, A.S. Sistemas Operacionais Modernos. 2. ed., Pearson Prentice Hall, 2008.</p> <p>SILBERSCHATZ, A.; GALVIN, P. B.; Gagne, G. Fundamentos de sistemas operacionais. Trad. 6. ed. LTC, 2009.</p> <p>TANENBAUM, A. S.; WOODHULL, A.S. Operating systems: design and implementation. 3 ed. Pearson Prentice Hall, 2009.</p>
Bibliografia Complementar	<p>STALLINGS, W. <i>Operating System: Internals and Design Principles</i>. 6. ed., Prentice Hall, 2008. ISBN-10: 0136006329, ISBN-13: 978-0136006329.</p> <p>MACHADO, F.B., MAIA, L.P. Arquitetura de Sistemas Operacionais. 4. ed., LTC, 2007. ISBN: 8521615485, ISBN-13: 9788521615484.</p> <p>DEITEL, H.M.; DEITEL, P.J. ; CHOHNES. Sistemas Operacionais. PRENTICE HALL BRASIL, 2007. ISBN: 8576050110, ISBN-13: 9788576050117.</p> <p>GUIMARAES, C. C. Princípios de sistemas operacionais. Rio de Janeiro: Campus, 1980. 222 p.</p> <p>KIRNER, C.; MENDES, S. B. T. Sistemas operacionais distribuídos: aspectos gerais e análise de sua estrutura. Rio de Janeiro: Campus, 1988. 184 p. ISBN 85-7001-475-</p>

Título	Computação e Sociedade
Objetivo Geral	Trabalhar temáticas relacionadas a problemas reais da sociedade nos quais a computação pode ter impactos positivos e negativos. Desenvolver proativamente as competências previstas para o egresso relacionadas ao senso crítico, à responsabilidade social, às questões étnico-raciais e de direitos humanos, entre outras. Familiarizar o estudante com os Objetivos de Desenvolvimento Sustentável (ODS) definidos pela ONU para enfrentamento de questões nas quais a computação pode colaborar.
Pré-requisitos	—
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Extensionistas: 60h
Caráter	Obrigatório
Ementa	
A disciplina será baseada em uma construção colaborativa de conhecimento com a disponibilização de um produto final à sociedade, em temática a ser definida em cada oferta, respeitando os objetivos. O tema será trabalhado sempre vinculado à computação. Exemplos de temas passíveis de serem trabalhados são: os ODS; aspectos profissionais, legais e éticos da computação; relações humanas, étnico-raciais, bullying, assédio moral e sexual; história e cultura afro-brasileira, africana e indígena; viés e discriminação algorítmicos; regulamentações brasileiras e estrangeiras. Exemplos de produtos que poderão ser entregues à sociedade são: produtos tecnológicos (sistemas e aplicativos computacionais), produtos de divulgação científica que visem democratizar o acesso ao conhecimento (materiais em redes sociais), bem como ações diretas de inclusão (cursos ofertados à sociedade). A disciplina deve identificar demandas e necessidades junto à sociedade.	
Respons. pela oferta	DC (NOVA)
Bibliografia Básica	
Organização das Nações Unidas. Transformando nosso mundo: A Agenda 2030 para o Desenvolvimento Sustentável. Resolução A/RES/70/1 da Assembleia Geral da ONU, 2015. Disponível em: https://brasil.un.org/sites/default/files/2020-09/agenda2030-pt-br.pdf .	
Extensão universitária: diálogos e possibilidades. (Orgs.) Eliane Garcia Rezende, Elisângela Monteiro Pereira, Vânia Reghina Bressan. Alfenas, MG : Unifal, 2020. 260 p. ISBN 9788563473257.	
LEMOS, R. Direito, tecnologia e cultura. Rio de Janeiro: Ed. FGV, 2005. 211 p. ISBN 85-225-0516-0.	
COLBARI, A.de L. Ética do trabalho: a vida familiar na construção da identidade profissional. São Paulo: Letras & Letras, 1995. 278 p. ISBN 85-85387-53-X	
GUERIN, B. Analyzing social behavior: behavior analysis and the social sciences. Reno: Context Press, 1994. 382 p. ISBN 1-87978-13-6.	
FREITAS, M. E.; HELOANI, J. R. M; BARRETO, M. Assédio moral no trabalho. São Paulo: Cengage Learning, 2013. 124 p. ISBN 9788522106288.	

Bibliografia Complementar

- SCIENTIFIC authorship: credit and intellectual property in science.** New York: Routledge, 2003. 384 p. ISBN 0-415-94293-4.
- SENNETT, R. A corrosão do caráter.** 4. ed. Rio de Janeiro: Record, 2006. 11. ed. 204 p. ISBN 85-01-0561-5.
- BOFF, L. Sustentabilidade:** o que é, o que não é. Petrópolis: Vozes, 2013. 2. ed. 200 p. ISBN 9788532642981.
- BAUMGARTEN, M. Conhecimento e sustentabilidade:** políticas de ciência, tecnologia e inovação no Brasil contemporâneo. Porto Alegre: Ed. da UFRGS, 2008. 262 p. ISBN 9788538600008.
- PICKERING, Peg. Como administrar conflitos profissionais: técnicas para transformar conflitos em resultados.** 10. ed. São Paulo: Market Books, 1999. 114 p. ISBN 85-87393-28-6.
- VIGNATTI, A. L. A Máquina da Natureza:** Uma perspectiva cronológica da Ciência da Computação teórica. S.l.: Clube de Autores, 2024. 425 p. ISBN 978-6501105260.
-

Título	Metodologia Científica
Objetivo Geral	Habilitar o estudante a compreender e dominar os mecanismos do processo de investigação científica tanto para o desenvolvimento de uma Iniciação Científica (IC) e do Trabalho de Conclusão de Curso (TCC) quanto para sua atuação profissional. Familiarizar o estudante com a metodologia do trabalho científico caracterizando procedimentos básicos, pesquisa bibliográfica, projetos e relatórios; publicações e trabalhos científicos; e os princípios e práticas para a elaboração de documentos científicos (artigos, relatórios) e do TCC.
Pré-requisitos	—
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Teóricas: 60h
Caráter	Obrigatório
Ementa	Caracterização do que é pesquisa, sua motivação e metodologia de desenvolvimento. Apresentação dos tipos de pesquisa (iniciação científica, trabalho de conclusão de curso etc.) e seus objetivos. Introdução aos principais conceitos relacionados à pesquisa (como objetivo, tema, problema, hipótese e justificativa). Descrição detalhada das etapas da pesquisa: determinação do tema-problema de trabalho, revisão bibliográfica, construção lógica do trabalho, desenvolvimento do trabalho e redação do texto. Conceituação de aspectos da ética na pesquisa científica: definição, princípios, plágio, conduta ética na pesquisa científica. Aprofundamento da organização da escrita científica: estrutura formal do trabalho, suas partes e conteúdo esperado, tipos de publicações científicas e suas peculiaridades. Orientação sobre a elaboração de referências e citações bibliográficas e a apresentação da pesquisa.
Respons. pela oferta	DC (NOVA)
Bibliografia Básica	<p>WAZLAWICK, R. S. Metodologia de pesquisa para ciência da computação. Rio de Janeiro: Elsevier, 2009. 159 p.</p> <p>CERVO, A. L.; BERVIAN, P. A.; da Silva, R. Metodologia científica. 4. ed. São Paulo: Makron, 1996. 209 p.</p> <p>LAKATOS, E. M.; MARCONI, M. de A. Fundamentos de metodologia científica. 7. ed. São Paulo: Atlas, 2010. 297 p. ISBN 978-85-224-5758-8.</p>
Bibliografia Complementar	<p>CASELI, H. UFSCAR. SEAD. Metodologia científica. São Carlos, SP: EdUFSCar, 2013.</p> <p>BARROS, A. J. da S.; LEHFELD, N. A. de S. Fundamentos de metodologia científica. 3. ed. São Paulo: Pearson Prentice Hall, 2007. 158 p. ISBN 978-85-7605-156-5.</p> <p>BASTOS, C. L.; KELLER, V. Aprendendo a aprender: introdução a metodologia científica. 18. ed. Petrópolis: Vozes, 1998. 111 p. ISBN 85-326-0586-9.</p> <p>PARRA FILHO, D.; SANTOS, J. A. Metodologia científica. 4. ed. São Paulo: Ed. Futura, 2001. 277 p. ISBN 85-86082-81-3.</p> <p>KOCHE, J. C. Fundamentos de metodologia científica: teoria da ciência e iniciação à pesquisa. 29. ed. Petrópolis, RJ: Vozes, 2011. 182 p. ISBN 9788532618047.</p> <p>MATTAR, J. Metodologia científica na era da informática. 3. ed. São Paulo: Saraiva, 2011. 308 p. ISBN 9788502064478.</p> <p>SANTOS, J. A.; PARRA FILHO, D. Metodologia científica. 2. ed. São Paulo: Cengage Learning, 2012. 251 p. ISBN 9788522112142.</p>

4.5.4 Disciplinas do Quarto Semestre

Título	Álgebra Linear 1
Objetivo Geral	A disciplina tem por objetivo levar os alunos a: interagir com fontes diretas e indiretas, selecionando e examinando criticamente tais fontes de modo a conduzir a uma prática de aprendizado continuado e autônomo em Álgebra Linear; dominar conhecimentos e habilidades de Álgebra Linear relacionando esses conhecimentos e habilidades com áreas correlatas como física e engenharia, através da modelagem, resolução e análise de tais modelos; criar e demonstrar resultados simples em Álgebra Linear e áreas correlatas; reconhecer a existência de características típicas de Álgebra Linear (vetores, bases, normas, sistemas lineares etc) em problemas e as utilizar adequadamente.
Pré-requisitos	Geometria Analítica
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Teóricas: 45h ▷ Práticas: 15h
Caráter	Obrigatório
Ementa	Estudo dos fundamentos de espaços vetoriais, bases e dimensão. Caracterização de subespaços vetoriais e sua decomposição em soma direta. Análise e interpretação geométrica de transformações lineares, casos especiais como rotações, dilatações, projeções, e suas aplicações. Desenvolvimento da representação matricial de transformações lineares e de mudança de base. Investigação de autovalores e autovetores e a diagonalização de matrizes, casos especiais como as matrizes simétricas/auto-adjuntas e suas aplicações. Fundamentos de espaços com produto interno, normas, ângulos e medidas de correlação entre vetores. Desenvolvimento do processo de ortogonalização de Gram-Schmidt. Estudo de formas bilineares, sua representação matricial, e suas aplicações, inclusive para a classificação de pontos críticos de funções de várias variáveis.
Respons. pela oferta	DM (1003501)
Bibliografia Básica	
	BOLDRINI, J. L. et al. Álgebra Linear . 3. ed. São Paulo: Harbra, 1986.
	CALLIOLI, C. A.; DOMINGUES, H. H.; COSTA, R. C. F. Álgebra linear e aplicações . 6. ed. São Paulo: Atual, 2013.
	COELHO, F. U.; LOURENÇO, M. L. Um curso de álgebra linear . 2. ed. São Paulo: EdUSP, 2010.
Bibliografia Complementar	
	ANTON, H.; BUSBY, R. C. Álgebra linear contemporânea . Porto Alegre, RS: Bookman, 2008.
	LIMA, E. L. Álgebra linear . 5. ed. Rio de Janeiro: IMPA. (Coleção Matemática Universitária), 2001.
	HOFFMAN, K.; KUNZE, R. Álgebra linear . 2. ed. Rio de Janeiro: Livros Técnicos e Científicos, 1979.
	LANG, S. Álgebra linear . São Paulo: Edgard Blucher, 1971.
	LIPSCHUTZ, S. Álgebra linear . São Paulo: McGraw-Hill do Brasil, 1973.
	MONTEIRO, L. H. J. Álgebra linear . São Paulo: Nobel, 1970.

Título	Projeto e Análise de Algoritmos
Objetivo Geral	
A disciplina tem por objetivo promover nos estudantes versatilidade na construção de soluções algorítmicas para problemas computacionais avançados, bem como capacidade crítica para analisar pontos fortes e fracos destas soluções. Com isso, deve tornar os estudantes aptos a aplicar estratégias algorítmicas avançadas a seus projetos; capacitar os estudantes a analisar a correção e o desempenho de algoritmos não-triviais; permitir aos estudantes consolidar os paradigmas de projeto de algoritmos (divisão e conquista, aleatorização, guloso, programação dinâmica), através de diversos exemplos e demonstrações; familiarizar os estudantes com noções da teoria da complexidade computacional; estimular os estudantes a avaliar quais técnicas de projeto, algoritmos e estruturas de dados se adéquam melhor a cada situação, problema ou aplicação.	
Pré-requisitos	Algoritmos e Estruturas de Dados 2
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Teóricas: 60h
Caráter	Obrigatório
Ementa	
Detalhamento das análises assintóticas (notações O, Omega e Theta). Aprofundamento de divisão-e-conquista: árvore de recorrência e teorema mestre (demonstração, interpretação e exemplos). Apresentação de aplicações em áreas distintas com definição do problema, algoritmo, recorrência, análises de correção e eficiência. Exemplos de aplicações: multiplicações de inteiros e de matrizes, ordenação e seleção aleatorizados (revisão de probabilidade), encontrar par mais próximo. Revisão de grafos e apresentação do conceito de corte em grafos. Aprofundamento de algoritmos gulosos: aplicações em áreas distintas com definição do problema, algoritmo e invariantes, análises de correção e eficiência. Exemplos de aplicações: escalonamento de tarefas em uma única máquina, seleção de atividades, mochila fracionária, código de Huffman, caminhos mínimos sem custos negativos, árvore geradora mínima. Detalhamento dos algoritmos para árvore geradora mínima de Prim (com e sem <i>heap</i>) e de Kruskal (com detalhamento da estrutura <i>union-find</i>). Aprofundamento de programação dinâmica (PD): aplicações em áreas distintas com definição do problema, subestrutura ótima com demonstração, recorrência, algoritmo, implementação eficiente, análises de correção e eficiência. Exemplos de aplicações: conjunto independente em grafos caminhos, alinhamento de sequências, mochila binária, árvore de busca ótima, caminhos mínimos com custos negativos. Detalhamento de algoritmos para caminhos mínimos de Bellman-Ford, Floyd-Warshall ou Johnson. Introdução de NP-Completude pelo ponto de vista algorítmico: reduções, completude, definição e interpretação de NP-Completude, questão P vs NP. Noções de abordagens para tratar problemas NP-Completos e NP-Difíceis.	
Respons. pela oferta	DC (NOVA)
Bibliografia Básica	
DASGUPTA, S.; PAPADIMITRIOU, C. H.; VAZIRANI, U. V. Algoritmos . New York: McGraw-Hill, 2009.	
CORMEN, T. H.; LEISERSON, C. E.; RIVEST, R. L.; STEIN, C. Introduction to Algorithms . 3. ed. Boston: McGraw-Hill, 2009.	
SEGEWICK, R.; WAYNE, K. Algorithms . 4. ed. Boston: Addison-Wesley, 2011.	

Bibliografia Complementar

- KLEINBERG, J.; TARDOS, É. *Algorithm Design*. Boston: Addison-Wesley, 2005.
- SEGEWICK, R. *Algorithms in C++, Parts 1-4: fundamentals, data structures, sorting, searching*. 3. ed. Boston: Addison-Wesley, 1998.
- SEGEWICK, R. *Algorithms in C++, Part 5: graph algorithms*. 3. ed. Boston: Addison-Wesley, 2001.
- ZIVIANI, N. **Projeto de algoritmos: com implementações em Java e C++**. 2. ed. São Paulo: Cengage Learning, 2011.
- KNUTH, D. E. *The Art of Computer Programming*. v. 1 e 3. Boston: Addison-Wesley, 1973.
- ROSEN, K. H. *Discrete mathematics and its applications*. 7. ed. New York: McGraw-Hill, 2013.
-

Título	Interação Humano-Computador
Objetivo Geral	Tornar os estudantes aptos a considerar requisitos de usuário e aspectos de qualidade de uso na construção de sistemas computacionais interativos; capacitar os estudantes a fazer design de sistemas computacionais interativos, adotando modelos e técnicas bem estabelecidos; capacitar os estudantes a realizar avaliações de sistemas computacionais interativos, adotando modelos e técnicas bem estabelecidos.
Pré-requisitos	Construção de Algoritmos e Programação
Disc. recomendadas	—
Carga horária total	60h
Natureza	► Teóricas: 30h ► Práticas: 30h
Caráter	Obrigatório
Ementa	Visão geral da Interação Humano-Computador: histórico, áreas e disciplinas envolvidas. Apresentação do conceito de sistemas computacionais interativos. Apresentação de fundamentos teóricos: fatores humanos e ergonomia, modelos de engenharia, conceitos de qualidade de uso. Aprofundamento em design de sistemas computacionais interativos: abordagens ao design, modelagem da interação, apoio a decisões de design, técnicas e estilos de prototipação, documentação de decisões de design. Aprofundamento em avaliação de sistemas computacionais interativos: avaliação analítica e empírica, métodos e técnicas de avaliação de usabilidade e acessibilidade.
Respons. pela oferta	DC (1001508)
Bibliografia Básica	
ROGERS, Y.; PREECE, J.; SHARP, H. Design de interação: além da interação homem-computador. 3. ed. Porto Alegre, RS: Bookman, 2013. xiv, 585 p. : il. (color.) ISBN 9788582600061.	
BARBOSA, S. D. J.; SILVA, B. S. da. Interação humano-computador. Rio de Janeiro: Elsevier, 2010. 384 p. ISBN 9788535234183.	
TULLIS, T.; ALBERT, B. <i>Measuring the user experience: collecting, analyzing, and presenting usability metrics.</i> Burlington: Elsevier, 2008. 317 p. (The Morgan Kaufmann Series in Interactive Technologies). ISBN 978-0-12-373558-4.	
Bibliografia Complementar	
ROCHA, H.; BARANAUSKAS, M. C. C. Design e Avaliação de Interfaces Humano-Computador. São Paulo - Escola Computação: IME - USP, 2003. v.1. 242p. ISBN 85-88833-04-2	
DIX, A.; FINLAY, J.; ABOWD, G.; BEALE, R. Human-Computer Interaction. 3. ed. Pearson, 2004. ISBN 978-0130461094	
SHNEIDERMAN, B.; PLAISANT, C. Designing the User Interface: Strategies for Effective Human-Computer Interaction. 5. ed. Pearson Addison-Wesley, 2009. ISBN 978-0321537355.	

Título	Engenharia de Software 1
Objetivo Geral	Capacitar os estudantes a realizar levantamento de requisitos; capacitar os estudantes a elaborar modelos (diagramas) que traduzem os requisitos em uma solução de software de qualidade; tornar os estudantes aptos a especificar diagramas que cobrem vários níveis de abstração de um sistema de software; habilitar os estudantes a refletir sobre a modelagem de sistemas não triviais, como de tempo real, embarcados, ferramentas, etc.
Pré-requisitos	Programação Orientada a Objetos
Disc. recomendadas	—
Carga horária total	60h
Natureza	► Teóricas: 60h
Caráter	Obrigatório
Ementa	Histórico da Engenharia de Software. Visão sobre Ciclo de Vida de Desenvolvimento de Sistemas de Software. Detalhamento do processo de gerenciamento de requisitos com ênfase na elicitação e especificação: documento de requisitos e casos de uso. Detalhamento do Processo de Conversão de Requisitos em Modelos Conceituais (Diagramas de Classes e Diagramas de Sequência do Sistema - DSS). Introdução à Modelagem comportamental: Diagramas de Estado em nível de análise. Introdução ao Projeto de Software. Detalhamento da conversão dos modelos de análise em Modelos de Projeto: Diagrama de Classes e de Pacotes (Subsistemas). Apresentação do conceito de modularização (agrupamento de classes que atendem a determinado critério). Conversão dos Modelos de Análise em Modelos Projeto: Diagramas de Sequência/Colaboração. Diagrama de Estados em nível de projeto. Utilização de Diagramas de Componentes para modularização do sistema. Utilização de diagramas de implantação.
Respons. pela oferta	DC (1001530)
Bibliografia Básica	
	SOMMERVILLE, I. <i>Engenharia de software</i> . 9. ed. São Paulo: Pearson Prentice Hall, 2011. 529 p. ISBN 97885793611081;
	PRESSMAN, R. S.; MAXIM, B. R. <i>Engenharia de software: uma abordagem profissional</i> . 8. ed. Porto Alegre: AMGH, 2016. 940 p. ISBN 9788580555332.
	FURLAN, J. D. <i>Modelagem de objetos através da UML - Unified Modeling Language</i> . São Paulo: Makron Books, 1998. 329 p. ISBN 85-346-0924-1.
	BLAHA, M.; RUMBAUGH, M. <i>Modelagem e projetos baseados em objetos com UML 2</i> . 2. ed. Rio de Janeiro: Elsevier, 2006. 496 p. ISBN 8535217533.
Bibliografia Complementar	
	HUMPHREY, W. S. <i>A discipline for software engineering</i> . Addison-Wesley, 1995. 789 p. (SEI Series in Software Engineering). ISBN 0-201-54610-8.
	PFLEGGER, S. L. <i>Engenharia de software: teoria e prática</i> . 2. ed. São Paulo: Prentice Hall, 2004. 537 p. ISBN 85-87918-31-1.
	<i>ENGINEERING and managing software requirements</i> . Berlin: Springer, 2006. AURUM, Aybüke; WOHLIN, C.(Eds.), 478 p. (Institute for nonlinear science). ISBN 3-540-25043-3.

Título	Banco de Dados
Objetivo Geral	Familiarizar os estudantes com os conceitos fundamentais sobre banco de dados; capacitar os estudantes para a realização de projetos de banco de dados; habilitar os estudantes para o desenvolvimento de sistemas de banco de dados; tornar os estudantes aptos a desenvolver um sistema de banco de dados utilizando um sistema gerenciador de banco de dados relacional.
Pré-requisitos	Algoritmos e Estrutura de Dados 1
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Teóricas: 30h ▷ Práticas: 30h
Caráter	Obrigatório
Ementa	Conceitos básicos de banco de dados: arquitetura de um sistema de banco de dados; componentes de um sistema gerenciador de banco de dados, arquitetura cliente-servidor de banco de dados, modelos e esquemas de banco de dados. Projeto conceitual de banco de dados: modelo entidade-relacionamento e modelo entidade-relacionamento estendido. Projeto lógico de banco de dados: modelo relacional e mapeamento entre esquemas do nível conceitual para o nível lógico. Álgebra relacional. Linguagem SQL.
Respons. pela oferta	DC (1001493)
Bibliografia Básica	
ELMASRI, R.; NAVATHE, S. B. Sistemas de banco de dados . 6. ed. São Paulo: Pearson Addison Wesley, 2011. 788 p. ISBN 9788579360855.	
RAMAKRISHNAN, R.; GEHRKE, J. Sistemas de gerenciamento de banco de dados . 3. ed. São Paulo: McGraw-Hill, 2008. 884 p. ISBN 978-85-7726-027-0.	
SILBERSCHATZ, A.; KORTH, H. F.; SUDARSHAN, S. Sistema de bancos de dados . 6. ed. São Paulo: Elsevier, 2012. 861 p. ISBN 978-85-352-4535-6.	
Bibliografia Complementar	
DATE, C. J. Introdução a sistemas de banco de dados . Rio de Janeiro: Elsevier, 2003. 865 p. ISBN 9788535212730.	
HEUSER, C. A. Projeto de banco de dados . 6. ed. Porto Alegre, RS: Bookman, 2009. 282 p. (Série Livros Didáticos Informática UFRGS ; v.4). ISBN 9788577803828.	
GARCIA-MOLINA, H.; ULLMAN, J. D.; WIDOM, J. Database system implementation . New Jersey: Prentice Hall, 2000. 653 p. ISBN 0-13-040264-8.	

Título	Inteligência Artificial
Objetivo Geral	Capacitar o estudante para utilizar representação de conhecimento na construção de algoritmos a partir dos conceitos da IA. Propiciar ao estudante a aquisição dos conceitos relacionados à busca, representação de conhecimento, raciocínio automático e aprendizado de máquina. Desenvolver no estudante a competência para saber identificar problemas que podem ser resolvidos com técnicas da IA e quais técnicas podem ser adequadas a cada problema.
Pré-requisitos	Algoritmos e Estruturas de Dados 1
Disc. recomendadas	Estatística Básica, Algoritmos e Estruturas de Dados 2
Carga horária total	60h
Natureza	► Teóricas: 30h ► Práticas: 30h
Caráter	Obrigatório
Ementa	Caracterização da área de IA. Apresentação de métodos de busca desinformada e informada para a resolução de problemas: busca em largura, busca de custo uniforme, busca em profundidade, subida da encosta, têmpera simulada, algoritmos evolutivos. Introdução à representação de conhecimento baseada em lógica. Visão geral de métodos de raciocínio e inferência: algoritmos de encadeamento para frente e para trás, resolução e programação lógica. Introdução à representação de conhecimento incerto: quantificação de incerteza e raciocínio probabilístico. Noções de aprendizado de máquina supervisionado e não-supervisionado: classificação, regressão e agrupamento.
Respons. pela oferta	DC (1001336)
Bibliografia Básica	
RUSSELL, S. J; NORVIG, P. <i>Artificial intelligence: a modern approach</i> . 3. ed. Upper Saddle River: Prentice-Hall, c2010. 1131 p. ISBN 978-0-13-604259-4.	
LUGER, G. F. <i>Artificial intelligence: Structures and strategies for complex problem solving</i> . 5. ed. Harlow: Addison Wesley Longman, c2005. 824 p. ISBN 0-321-26318-9.	
BRATKO, I. <i>Prolog: programming for artificial intelligence</i> . 2. ed. Harlow: Addison-Wesley, 1990. 597 p. (International Computer Science Series). ISBN 0-201-41606-9.	
Bibliografia Complementar	
MITCHELL, T. M. <i>Machine learning</i> . Boston: MCB/McGraw-Hill, 1997. 414 p. (McGraw-Hill Series in Computer Science). ISBN 0-07-042807-7;	
BITTENCOURT, G. <i>Inteligência artificial: ferramentas e teorias</i> . 3. ed. Florianópolis, SC: Editora da UFSC, 2006. 3p. 371 (Série Didática). ISBN 8532801382;	
FACELI, K.; LORENA, A. C.; GAMA, J.; CARVALHO, A. C. P. de L. F. de. <i>Inteligência artificial: uma abordagem de aprendizado de máquina</i> . Rio de Janeiro: LTC, 2011. 378 p. ISBN 9788521618805;	
COPPIN, B. <i>Inteligência artificial</i> . Grupo Gen-LTC, 2015.	

4.5.5 Disciplinas do Quinto Semestre

Título	Cálculo Numérico
Objetivo Geral	A disciplina tem por objetivo levar os alunos a: interagir com fontes diretas e indiretas, selecionando e examinando criticamente tais fontes de modo a conduzir a uma prática de aprendizado continuado e autônomo em Cálculo Numérico; dominar conhecimentos e habilidades de Cálculo Numérico relacionando esses conhecimentos e habilidades com áreas correlatas como física e engenharia, através da modelagem, resolução e análise de tais modelos; criar e demonstrar resultados simples em Cálculo, Álgebra e áreas correlatas sob o ponto de vista de aproximações; reconhecer a existência de características típicas de Cálculo Numérico (erros, aproximações polinomiais de funções, zeros de funções, etc) em problemas e as utilizar adequadamente.
Pré-requisitos	Cálculo Diferencial e Integral 1, Geometria Analítica e Construção de Algoritmos e Programação
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Teóricas: 45h ▷ Práticas: 15h
Caráter	Obrigatório
Ementa	Estudo da aritmética de ponto flutuante e análise de erros em operações computacionais. Desenvolvimento de métodos iterativos e diretos para resolução de sistemas lineares, incluindo o método de Gauss-Jacobi e técnicas de fatoração matricial como LU e Cholesky, com análise de custo computacional. Investigação de métodos numéricos para resolução de equações não-lineares, como os métodos Newton, secante e da bissecção; suas propriedades de convergência e critérios de parada. Estabelecimento das técnicas de interpolação polinomial para pontos arbitrariamente espaçados, com análise do erro de interpolação e sua relação com a escolha dos pontos e grau do polinômio. Aplicação do método dos mínimos quadrados no ajuste de curvas a dados experimentais, incluindo transformações logarítmicas. Desenvolvimento de fórmulas de diferenças finitas para aproximação de derivadas, com análise da influência do espaçamento entre pontos nos erros de truncamento e arredondamento. Estudo de métodos de integração numérica, com foco na ordem do método e sua relação com a precisão desejada. Investigação dos métodos numéricos para equações diferenciais ordinárias, incluindo os métodos de Euler e Runge-Kutta, com análise da relação entre ordem do método e eficiência computacional em simulações numéricas de problemas aplicados.
Respons. pela oferta	DM (1003726)
Bibliografia Básica	<p>RUGGIERO, M.; LOPES, V. L. Cálculo Numérico: aspectos teóricos e computacionais, MacGraw-Hill, 1996.</p> <p>ARENALES, S.; DAREZZO, A. Cálculo Numérico: Aprendizagem com apoio de software, Editora Thomson, 2007.</p> <p>FRANCO, N. B. Cálculo Numérico. Pearson Prentice Hall, 2006.</p>

Bibliografia Complementar

- HUMES et al. **Noções de Cálculo Numérico**. MacGraw-Hill, 1984.
- BARROSO, C. L. et al. **Cálculo Numérico com Aplicações**. Harbra, 1987.
- BURDEN, R.L.,FAIRES, J.D. **Numerical Analysis**. PWS Publishing Company, 1996.
- CLÁUDIO, D. M. et al. **Fundamentos de Matemática Computacional**. Atlas, 1989).
- CONTE, S. D. **Elementos de Análise Numérica**. Ed. Globo, 1975.
- DEMIDOVICH, B. P. et al. **Computacional Mathematics**. Moscou, Mir Pub, 1987.
- SANTOS, V. R. **Curso de Cálculo Numérico**. LTC, 1977.
- SPERANDIO, D. et al. **Cálculo Numérico: características matemáticas e computacionais dos métodos numéricos**. Pearson/Prentice Hall, 2003.
- YONG, D. M. et al. **Survey of Numerical Mathematics**. Addison Wesley, 1972.
-

Título	Redes de Computadores
Objetivo Geral	Estudar as redes de computadores, abordando suas operações, funcionalidades e serviços. Apresentar tecnologias de conexão existentes, abordando aspectos de hardware e de protocolos e o projeto físico e lógico de redes.
Pré-requisitos	Sistemas Operacionais
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Teóricas: 60h
Caráter	Obrigatório
Ementa	Transmissão de dados: camadas física e de enlace, sinalização, modulação e codificação, framing, endereçamento, camadas física e de enlace. Endereçamento lógico e físico, encaminhamento, roteamento e mobilidade na Internet. Endereçamento físico e lógico, roteamento fixo e dinâmico, mobilidade de nós, encaminhamento de pacotes. Controle de fluxo e de congestionamento: latência, bufferbloat, bandwidth, throughput, controle de fluxo fim a fim, controle de congestionamento na rede. Gerenciamento de rede: configuração, desempenho, contabilização, falha e segurança. Redes definidas por software (SDN), Redes de sensores, redes móveis, redes ad-hoc e redes veiculares. Qualidade de Serviço (QoS).
Respons. pela oferta	DC (1001504)
Bibliografia Básica	
TANENBAUAN, A. <i>Computer Networks</i> . Prentice-Hall, 3. ed., 1996.	
KUROSE, J. F. ; ROSS, K. W. <i>Redes de Computadores e a Internet: Uma abordagem top-down</i> . Pearson Addison Wesley– 6 ^a Edição, 2014	
PETERSON, L. L.; DAVIE, B. S. <i>Computer Networks: A Systems Approach</i> , 5. ed., Editora Elsevier.	
Bibliografia Complementar	
COMER, D. E. <i>Redes de Computadores e Internet</i> . 6. ed. Editora Bookman, 2016	

Título	Projeto e Análise de Algoritmos 2
Objetivo Geral	A disciplina tem por objetivo promover nos estudantes versatilidade na construção de soluções algorítmicas para problemas computacionais particularmente complexos e difíceis, bem como capacidade crítica para analisar pontos fortes e fracos destas soluções. Para tanto, deve apresentar aos estudantes alguns dos problemas computacionais mais desafiadores da classe P, como fluxo em redes, emparelhamento e programação linear, junto de soluções algorítmicas eficientes para os mesmos. Deve também apresentar problemas NP-difíceis clássicos, junto de abordagens algorítmicas variadas e complementares para os mesmos, incluindo métodos exatos, heurísticos e aproximados.
Pré-requisitos	Projeto e Análise de Algoritmos
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Teóricas: 60h
Caráter	Optativo
Ementa	Detalhamento dos problemas de fluxo máximo e corte mínimo, incluindo dualidade, grafo residual, caminhos aumentadores, algoritmo pseudo polinomial, algoritmos polinomiais, e aplicações. Apresentação dos problemas de emparelhamento estável e emparelhamento em grafos bipartidos, junto de algoritmos eficientes para os mesmos. Detalhamento de Programação Linear, incluindo dualidade e algoritmo simplex. Apresentação de reduções entre problemas, tanto para a obtenção de novos algoritmos quanto para demonstrar a NP-dificuldade de um problema. Introdução a otimização combinatória com apresentação de problemas ND-difíceis clássicos, como Escalonamento, Mochila Binária, Coloração, Corte Máximo, Caixeiro Viajante, Localização de Instalações, Cobertura por Vértices, Cobertura por Conjuntos, Árvore de Steiner e Conjunto Independente. Apresentação de métodos exatos para resolução de problemas NP-difíceis, com detalhamento de <i>Branch-and-Bound</i> , Programação Linear Inteira e Programação por Restrições, incluindo modelagem de problemas e funcionamento dos resolutores. Apresentação de abordagens heurísticas para resolver problemas NP-difíceis, incluindo algoritmos construtivos e de busca local. Detalhamento de algoritmos de aproximação para problemas NP-difíceis, incluindo diferentes abordagens como arredondamento de Programação Linear e método primal-dual.
Respons. pela oferta	DC (NOVA)
Bibliografia Básica	
DASGUPTA, S.; PAPADIMITRIOU, C. H.; VAZIRANI, U. V. Algoritmos . New York: McGraw-Hill, 2009.	
KLEINBERG, J.; TARDOS, É. Algorithm Design . Boston: Addison-Wesley, 2005.	
WILLIAMSON, D. P. Network Flow Algorithms , Cambridge University Press, 2019.	
WILLIAMSON, D. P.; SHMOYS, D. B. The Design of Approximation Algorithms , Cambridge University Press, 2011.	
CORMEN, T. H.; LEISERSON, C. E.; RIVEST, R. L.; STEIN, C. Introduction to Algorithms . 3. ed. Boston: McGraw-Hill, 2009.	
Bibliografia Complementar	
SEGEWICK, R.; WAYNE, K. Algorithms . 4. ed. Boston: Addison-Wesley, 2011.	
VAZIRANI, V. V. Approximation Algorithms , Berlin: Springer, 2001.	
ROSEN, K. H. Discrete mathematics and its applications . 7. ed. New York: McGraw-Hill, 2013.	

Título	Programação Orientada a Objetos Avançada
Objetivo Geral	Estimular o estudante a programar utilizando estruturas que facilitem a implementação, manutenção e evolução de software. Familiarizar o estudante com os princípios SOLID (responsabilidade única, aberto-fechado, substituição de Liskov, segregação de interface e inversão de dependência) da orientação a objetos. Capacitar o estudante a criar software orientado a objetos que utiliza os conceitos básicos da programação orientada a objetos (abstração, classes, objetos, atributos e métodos, encapsulamento/visibilidade, herança, composição/agregação, sobrecarga, polimorfismo de inclusão, classes abstratas, polimorfismo paramétrico, modularização, alocação dinâmica de objetos, tratamento de exceções) de forma a corretamente seguir os princípios SOLID.
Pré-requisitos	Algoritmos e Estruturas de Dados 1 e Programação Orientada a Objetos
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Teóricas: 30h ▷ Práticas: 30h
Caráter	Optativo
Ementa	Histórico da orientação a objetos. Princípios SOLID (responsabilidade única, aberto-fechado, substituição de Liskov, segregação de interface e inversão de dependência). Atribuição de responsabilidade em programas orientados a objetos. Padrões de projeto em nível de implementação. Prática de desenvolvimento de software orientado a objetos seguindo os princípios SOLID.
Respons. pela oferta	DC (1001521)
Bibliografia Básica	<p>LARMAN, C. Utilizando UML e Padrões: uma introdução à análise e ao projeto orientados a objetos e ao desenvolvimento iterativo. 3. ed. Porto Alegre: Bookman, 2006. 696 p. ISBN 8560031529.</p> <p>WEST, D. Use a Cabeça! Análise & Projeto Orientado ao Objeto. Rio de Janeiro: Alta Books, 2007. 472 p. ISBN 978-85-7608-145-6.</p> <p>GAMMA, E.; HELM, R.; VLISSIDES, J.; JOHNSON, R. Padrões de Projeto: soluções reutilizáveis de software orientado a objetos. Porto Alegre: Bookman, 2003. 368 p. ISBN 8573076100.</p>
Bibliografia Complementar	<p>DEITEL, H.M. & DEITEL, P. J. C++ Como Programar. 5. ed. Pearson Prentice Hall, 2006</p> <p>PIZZOLATO, E. B. Introdução à programação orientada a objetos com C++ e Java. EdUFSCar, 2010</p> <p>ECKEL, B. Thinking in C++. 2. ed. Upper Saddle River: Prentice Hall, 2000.</p> <p>SILVA FILHO, A. M. Introdução à programação orientada a objetos com C++. Elsevier, 2010</p> <p>DEITEL, Paul J.; DEITEL, Harvey M. C++ for programmers. Upper Saddle River, NJ: Prentice Hall, 2009. 1000 p. (Deitel Developer Series). ISBN 10-0-13-700130-9.</p> <p>SCHILDT, Herbert. C++: the complete reference. 4. ed. New York: McGraw Hill, c2003. 1023 p. ISBN 0-07-222680-3.</p>

Título	Desenvolvimento de Software para Web 1
Objetivo Geral	Familiarizar o estudante com os principais conceitos do desenvolvimento de software para web; capacitar o estudante a desenvolver aplicações web pelo lado do servidor (back-end).
Pré-requisitos	Banco de Dados
Disc. recomendadas	—
Carga horária total	60h
Natureza	► Práticas: 60h
Caráter	Optativo
Ementa	Conceitos de requisição/resposta. Navegação entre recursos web (redirecionamento, encaminhamento e inclusão). Compartilhamento de informações em nível de requisição, sessão e contexto. Geração de conteúdo dinâmico no servidor. Padrões arquiteturais para web. Frameworks para desenvolvimento Web.
Respons. pela oferta	DC (1001542)
Bibliografia Básica	
Oracle. Java Platform, Enterprise Edition: The Java EE Tutorial. Disponível em: https://docs.oracle.com/javaee/	
BASHAM, B. B.; SIERRA, K. Use a Cabeça! Servlets & JSP. 2. ed. Alta Books. 2009.	
LUCKOW, D. H.n; de MELO, A. A. Programação Java para a Web. 2. ed., Novatec, 2015. 680p.	
Bibliografia Complementar	
DEITEL, H.; DEITEL, P. Java: como programar. 8 edição. São Paulo: Prentice Hall, 2005. (https://plataforma.bvirtual.com.br/Leitor/Publicacao/39590/pdf)	
HALL, Marty; BROWN, Larry. Core web programming. 2. ed. Upper Saddle River: Prentice Hall, 2001.	
MOREIRA NETO, Oziel. Entendendo e dominando o Java para internet: desenvolva aplicativos Java para a web. São Paulo: Digerati Books, 2006.	

Título	Engenharia de Software 2
Objetivo Geral	Habilitar o estudante a gerenciar o processo de desenvolvimento de um sistema de software; habilitar o estudante a aplicar testes funcionais e estruturais em sistemas de software; familiarizar o estudante com conceitos de qualidade de software e fazer com que ele consiga refletir esses conceitos na prática.
Pré-requisitos	Engenharia de Software 1
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Teóricas: 60h
Caráter	Optativo
Ementa	Aprofundamento sobre Ciclo de Vida de Desenvolvimento e Manutenção de Sistemas. Modelos de Processo e Metodologias Ágeis: características, diretrizes de escolha e simulação. Técnicas de gerenciamento de projetos de software (local e distribuído geograficamente): Gerenciamento de configuração e de versões. Técnicas de Verificação e Validação de Software: Testes Funcionais e Estruturais; Conceituação e Exemplificação de Tipos de Manutenção de Software; Caracterização de qualidade de software e seu emprego/manutenção ao longo das fases do desenvolvimento; métricas, smells e refatorações. Visão geral sobre modelos de melhoria de processo.
Respons. pela oferta	DC (1001516)
Bibliografia Básica	<p>PRESSMAN, R. S.; MAXIM, B. R. <i>Engenharia de software: uma abordagem profissional</i>. 8. ed. Porto Alegre: AMGH, 2016. 940 p. ISBN 9788580555332.</p> <p>SOMMERVILLE, I. Engenharia de software. 9. ed. São Paulo: Pearson Prentice Hall, 2011. 529 p. ISBN 97885793611081.</p> <p>DELAMARO, M. E.; MALDONADO, J.C.; JINO, M. Introdução ao teste de software. Rio de Janeiro: Elsevier: Campus, 2007, 394 p. ISBN 9788535226348.</p>
Bibliografia Complementar	<p>GORTON, I. <i>Essential Software Architecture</i>. Springer-Verlag, Germany, 2016. ISBN 9783642191763</p> <p>COHN, M. <i>Agile estimating and planning</i>. Upper Saddle River, NJ: Prentice Hall Professional Technical Reference, 2010. 330 p. (Robert C. Martin Series). ISBN 9780131479418.</p> <p>THE CAPABILITY maturity model: guidelines for improving the software process. Boston: Addison-Wesley, 2001. 441 p. (The SEI Series in Software Engineering). ISBN 0-201-54664-7.</p> <p>PFLEEEGER, S. L. Engenharia de software: teoria e prática. 2. ed. São Paulo: Prentice Hall, 2004. 537 p. ISBN 85-87918-31-1.</p>

Título	Projeto e Implementação de Banco de Dados
Objetivo Geral	Tornar os estudantes aptos a realizar projeto de bancos de dados, abrangendo as fases de projeto conceitual, lógico e físico; estimular os estudantes a desenvolver um sistema de banco de dados utilizando um Sistema Gerenciador de Banco de Dados de grande porte; tornar os estudantes aptos nas tarefas e procedimentos de um administrador de banco de dados.
Pré-requisitos	Banco de Dados
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Teóricas: 30h ▷ Práticas: 30h
Caráter	Optativo
Ementa	Apresentação da teoria de dependência funcional e normalização; Explicitação sobre organização física dos dados, índices, visões não materializadas e visões materializadas; Introdução sobre processamento de transações e controle de concorrência. Programação em banco de dados: procedimentos armazenados, gatilhos, funções e cursores; Introdução ao uso de linguagens de programação e acesso a banco de dados usando APIs (application programming interfaces); Introdução ao uso de recursos de programação de interface com os SGBDs.
Respons. pela oferta	DC (1001546)
Bibliografia Básica	
ELMASRI, R.; NAVATHE, S. B. Sistemas de banco de dados . 6. ed. São Paulo: Pearson Addison Wesley, 2011. 788 p. ISBN 9788579360855.	
RAMAKRISHNAN, R.; GEHRKE, J. Sistemas de gerenciamento de banco de dados . 3. ed. São Paulo: McGraw-Hill, 2008. 884 p. ISBN 978-85-7726-027-0.	
SILBERSCHATZ, A.; KORTH, Henry F.; SUDARSHAN, S. Sistema de bancos de dados . 6. ed. São Paulo: Elsevier, 2012. 861 p. ISBN 978-85-352-4535-6.	
Bibliografia Complementar	
DATE, C. J. Introdução a sistemas de banco de dados . Rio de Janeiro: Elsevier, 2003. 865 p. ISBN 9788535212730.	
HEUSER, C. A. Projeto de banco de dados . 6. ed. Porto Alegre, RS: Bookman, 2009. 282 p. (Série Livros Didáticos Informática UFRGS ; v.4). ISBN 9788577803828.	
GARCIA-MOLINA, Hector; ULLMAN, Jeffrey D.; WIDOM, Jennifer. Database system implementation . New Jersey: Prentice Hall, 2000. 653 p. ISBN 0-13-040264-8.	

Título	Aprendizado de Máquina 1
Objetivo Geral	Familiarizar o estudante com conceitos básicos e algoritmos de aprendizado de máquina supervisionado e não-supervisionado. Capacitar o estudante a identificar quais algoritmos de aprendizado de máquina e quais ferramentas podem ser adequados a cada problema. Capacitar o estudante a realizar a análise de resultados desses algoritmos.
Pré-requisitos	Inteligência Artificial e Estatística Básica
Disc. recomendadas	Projeto e Análise de Algoritmos
Carga horária total	60h
Natureza	▷ Teóricas: 60h
Caráter	Optativo
Ementa	Apresentação de conceitos básicos e exemplos de aplicação de Aprendizado de Máquina. Noções de ferramentas e linguagens apropriadas para AM. Visão geral sobre aprendizado supervisionado: classificação, regressão e seleção de modelos e generalização. Detalhamento sobre técnicas de avaliação e comparação de modelos de classificação. Visão geral sobre aprendizado não-supervisionado: agrupamento, aprendizado competitivo e regras de associação. Introdução a técnicas de pré-processamento e redução de dimensionalidade: seleção e transformação de atributos e pré-processamento de dados não estruturados.
Respons. pela oferta	DC (1001524)
Bibliografia Básica	<p>MITCHELL, T. M. <i>Machine learning</i>. Boston: MCB/McGraw-Hill, 1997. 414 p. (McGraw-Hill Series in Computer Science). ISBN 0-07-042807-7</p> <p>WITTEN, I. H.; FRANK, E. <i>Data mining: practical machine learning tools and techniques</i>. 2. ed. San Francisco: Elsevier, 2005. 524 p. (The Morgan Kaufmann Series in Data Management Systems). ISBN 0-12-088407-0.</p> <p>ALPAYDIN, E. <i>Introduction to machine learning</i>. Cambridge: MIT Press, 2004. 415 p. (Adaptive Computation and Machine Learning). ISBN 0-262-01211-1.</p>
Bibliografia Complementar	<p>FACELI, K; LORENA, A. C.; GAMA, J.; CARVALHO, A. C. P. de L. F. de. Inteligência artificial: uma abordagem de aprendizado de máquina. Rio de Janeiro: LTC, 2011. 378 p. ISBN 9788521618805.</p> <p>PANG-NING, T.; STEINBACH, M.; KUMAR, V. <i>Introduction data mining</i>. Boston: Pearson Education, c2006. 769 p. ISBN 0-321-32136-7.</p> <p>BISHOP, C. M. <i>Pattern recognition and machine learning</i>. New York: Springer, c2006. 738 p. (Information Science and Statistics). ISBN 978-0-387-31073-2.</p> <p>HAYKIN, S. S. <i>Neural networks and learning machines</i>. 3. ed. Upper Saddle River: Pearson Education, 2008. 906 p. ISBN 978-0-13-147139-9.</p> <p>SILVA, I. N. da; SPATTI, D. H.; FLAUZINO, R. A. Redes neurais artificiais: para engenharia e ciências aplicadas. São Paulo: Artliber, 2010. 399 p. ISBN 978-85-88098-53-4.</p>

Título	Computação Gráfica
Objetivo Geral	Familiarizar o estudante com os conceitos fundamentais da área; capacitar o estudante a compreender a organização e as funcionalidades de sistemas gráficos; capacitar o estudante a implementar abordagens básicas na solução de problemas em computação gráfica.
Pré-requisitos	Geometria Analítica e Algoritmos e Estruturas de Dados 1
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Teóricas: 30h ▷ Práticas: 30h
Caráter	Optativo
Ementa	Introdução à computação gráfica; apresentar os tipos de equipamentos e tecnologias atuais disponíveis em computação gráfica; algoritmos básicos: aspectos geométricos e transformações (problemática associada e algoritmos). Noções da teoria de cores. Aprofundamento em modelagem de objetos bidimensionais e tridimensionais. Apresentação de projeções planares. Aprofundamento em transformações de visualização, determinação de superfícies visíveis e técnicas de iluminação e sombreamento. Visão geral de programação com pacotes gráficos padrões. Noções de gerenciamento de eventos. Noções de animação.
Respons. pela oferta	DC (1001536)
Bibliografia Básica	
ANGEL, E.; Shreiner D. Interactive Computer Graphics: A Top-Down Approach With WebGL . 7. ed. Pearson 2014.	
FOLEY, J. et al. Computer graphics: principles and practice . 3. ed. Addison-Wesley Professional, 2013, 1264 p.	
SHREINER, D. et al. OpenGL Programming Guide: The Official Guide to Learning OpenGL, Version 4.3 . 8. ed. Addison-Wesley, 2013, 935 p.	
Bibliografia Complementar	
COHEN, M.; MANSOUR, I. OpenGL - Uma Abordagem Prática e Objetiva . São Paulo: Novatec, 2006. 486 p.	
AZEVEDO, E.; CONCI, A.. Computação Gráfica. Geração de Imagem - Volume 1- Teoria e Prática . Elsevier, 2003. 384 p	
VELHO, L.; GOMES J. M. Fundamentos da Computação Gráfica . Rio de Janeiro: IMPA, 2008.	

Título	Processamento Digital de Imagens
Objetivo Geral	Habilitar o estudante a aplicar técnicas de melhoramento e segmentação de imagens digitais; habilitar o estudante a realizar o pré-processamento de imagens para a subsequente aplicação de técnicas de aprendizado de máquina; capacitar o estudante a identificar as técnicas mais adequadas a serem aplicadas dependendo do tipo de imagem a ser processada.
Pré-requisitos	Construção de Algoritmos e Programação, Cálculo Diferencial e Integral 1 e Geometria Analítica
Disc. recomendadas	—
Carga horária total	60h
Natureza	► Teóricas: 30h ► Práticas: 30h
Caráter	Optativo
Ementa	Visão biológica e artificial. Visão geral das etapas de um sistema de processamento de imagens. Apresentação de técnicas de modificação de histogramas. Detalhamento sobre filtragem espacial de imagens (filtros lineares e não-lineares). Aprofundamento sobre filtragem de imagens no domínio da frequência. Apresentação de processamento multiresolução. Apresentação detalhada sobre processamento morfológico. Visão geral sobre técnicas de representação e descrição de imagens. Apresentação de técnicas de segmentação de imagens.
Respons. pela oferta	DC (1001527)
Bibliografia Básica	
GONZALEZ, R. C.; WOODS, R. E. <i>Digital Image Processing</i> . 3. ed. Upper Saddle River: Prentice-Hall, 2008.	
JAIN, A. K. <i>Fundamentals of Digital Image Processing</i> . Upper Saddle River: Prentice-Hall, 1989.	
PEDRINI, H.; ROBSON, W. <i>Análise de Imagens Digitais: Princípios, Algoritmos e Aplicações</i> . São Paulo: Thomson Learning, 2008.	
Bibliografia Complementar	
FORSYTH, D. A.; PONCE, J. <i>Computer Vision: A Modern Approach</i> . Upper Saddle River: Prentice Hall, 2003.	
SZELISKI, R. <i>Computer Vision: Algorithms and Applications</i> . New York: Springer, 2010. Disponível em: http://szeliski.org/Book/ .	
NIXON, M.; AGUADO, A. S. <i>Feature Extraction & Image Processing for Computer Vision</i> . 2. ed. London: Academic Press, 2008.	
OPENHEIM, A. V.; SCHAFER, R. W. <i>Discrete-Time Signal Processing</i> . Upper Saddle River: Prentice-Hall, 1989.	
PROAKIS, J. G.; MANOLAKIS, D. G. <i>Digital Signal Processing: Principles, Algorithms and Applications</i> . New York: MacMillan, 1992.	

Título	Sistemas Distribuídos
Objetivo Geral	Familiarizar o estudante com aspectos inerentes à interligação lógica de sistemas computacionais fracamente acoplados. Familiarizar o estudante com as dificuldades e técnicas para prover comunicação, sincronização e coordenação entre múltiplos sistemas de computação distribuídos. Capacitar o estudante a tratar do compartilhamento ordenado e seguro de recursos computacionais distribuídos. Capacitar o estudante a tratar do desenvolvimento de técnicas e infraestruturas de software para ambientes computacionais distribuídos. Habilitar o estudante a criar aplicações que usem de maneira eficiente múltiplos recursos computacionais distribuídos.
Pré-requisitos	Sistemas Operacionais
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Teóricas: 60h
Caráter	Optativo
Ementa	Motivações, objetivos e caracterização de Sistemas Distribuídos. Arquiteturas de sistemas distribuídos; middleware. Processos, threads e unidades de execução de código; modelos cliente / servidor e peer-to-peer; virtualização. Comunicação em rede, protocolos e APIs. Invocação de códigos remotos. Comunicação orientada a mensagens, a fluxos e multicast. Nomeação: identificadores e localização. Sincronização. Relógios físicos e lógicos. Ordenação. Exclusão mútua. Eleição; coordenação. Consistência e replicação: modelos de consistência; gerenciamento de réplicas; protocolos de consistência. Tolerância à falhas: modelos; redundância; resiliência de processos e de comunicação. Comunicação confiável. Acordos distribuídos e consenso; recuperação. Segurança: ameaças, políticas e mecanismos; criptografia; canais seguros; controle de acesso. Gerenciamento de segurança. Estudo de casos em Sistemas Distribuídos.
Respons. pela oferta	DC (1001503)
Bibliografia Básica	
TANENBAUM, A. S.; STEEN, M. V. Sistemas distribuídos: princípios e paradigmas . 2. ed. São Paulo. Pearson Prentice Hall, 2007.	
COULOURIS, G.; DOLLIMORE, J.; KINDBERG, T.; and BLAIR, G. Distributed systems: concepts and design . 5. ed. Addison-Wesley, 2012.	
TANENBAUM, A. S. Distributed operating systems . Prentice Hall, 1995. 614 p.	
Bibliografia Complementar	
Addison-Wesley, 2009. Ghosh, Sukumar. Distributed systems: an algorithmic approach . Chapman & Hall/CRC, c2007. BIRMAN, K. P. Reliable distributed systems: technologies, web services, and applications . New York: Springer, 2010. ANTONOPOULOS, N.; Gilliam L. Cloud computing: principles, systems and applications . New York. Springer, 2010. SINHA, P. K. Distributed operating systems: concepts and design . New York: IEEE Computer Society Press, 1997. TEL, G. Introduction to distributed algorithms . 2. ed. Cambridge University Press, 2000.	

Título	Processamento de Sinais Digitais
Objetivo Geral	Prover embasamento teórico do ferramental matemático básico para a análise de sinais e sistemas no tempo contínuo e discreto com exemplos de aplicação em problemas de engenharia.
Pré-requisitos	Cálculo Diferencial e Integral 1, Geometria Analítica, Álgebra Linear e Construção de Algoritmos e Programação
Disc. recomendadas	—
Carga horária total	90h
Natureza	▷ Teóricas: 60h ▷ Práticas: 30h
Caráter	Optativo
Ementa	Introdução ao processamento de sinais. Fundamentos matemáticos de sinais e sistemas. Convolução de sinais. Análise em frequência de sinais. Série de Fourier, Transformada de Fourier e transformada Z. Amostragem e reconstrução de sinais de tempo contínuo: Teorema de Nyquist e efeito de Aliasing. Filtros digitais: análise, estruturas, técnicas de projeto e aspectos práticos de implementação.
Respons. pela oferta	DC (1001486)
Bibliografia Básica	<p>OPPENHEIM, A.V., WILLSKY, A.S., NAWAB, S.H. <i>Signals and Systems</i>. 2. ed. Prentice Hall, 1997;</p> <p>OPPENHEIM, A.V., SCHAFER, R.W. <i>Discrete Time Signal Processing</i>. Prentice Hall, 1989;</p> <p>LATHI, B.P. <i>Sinais e sistemas lineares</i>. 2. ed. Porto Alegre, RS: Bookman, 2007. 856 p. ISBN 978-85-60031-13-9.</p>
Bibliografia Complementar	<p>SOLIMAN, S.S., SRINATH, M.D. <i>Continuous and Discrete Signals And Systems</i>. 2. ed., Prentice Hall, 1998.</p> <p>DENBIGH, P. <i>System Analysis & Signal Processing</i>. Addison Wesley, 1998.</p> <p>PROAKIS, J.G., MANOLAKIS, D.G. <i>Digital Signal Processing: Principles, Algorithms and Applications</i>. 4. ed., Pearson Prentice Hall, 2007.</p>

4.5.6 Disciplinas do Sexto Semestre

Título	Teoria da Computação
Objetivo Geral	Familiarizar os estudantes com a teoria de linguagens formais, a teoria de autômatos e a equivalência entre ambas. Capacitar os estudantes a descrever linguagens simples utilizando expressões regulares e gramaticais livres de contexto. Familiarizar os estudantes com noções de representação de problemas e soluções computacionais por meio dessas teorias. Tornar os estudantes aptos a reconhecer problemas indecidíveis e intratáveis por meio dessas teorias.
Pré-requisitos	Matemática Discreta
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Teóricas: 60h
Caráter	Obrigatório
Ementa	Introdução aos conceitos de alfabetos, gramáticas e linguagens; detalhamento das linguagens, expressões e gramáticas regulares. Apresentação dos autômatos finitos (máquinas de estados) e autômatos finitos com saída (máquinas de Mealy e Moore). Detalhamento das gramáticas e linguagens livres de contexto. Apresentação dos autômatos finitos com pilha. Aprofundamento em Máquinas de Turing. Hierarquia das classes de linguagens formais: gramáticas não-restritas e sensíveis ao contexto e linguagens recursivas e recursivamente enumeráveis. Aprofundamento nos limites da computação algorítmica: computabilidade e decidibilidade.
Respons. pela oferta	DC (1001510)
Bibliografia Básica	HOPCROFT, J. E.; MOTWANI, R.; ULLMAN, J. D. Introdução à Teoria de Autômatos, Linguagens e Computação . Rio de Janeiro: Editora Campus, 2003. SIPSER, M. Introdução à Teoria da Computação . 2. ed. norte-americana. São Paulo: Cengage CTP, 2007. 488 p. LEWIS, H. R.; PAPADIMITRIOU, C. H. Elementos de Teoria da Computação . 2. ed. Porto Alegre: Bookman, 2004.
Bibliografia Complementar	MENEZES, P. B. Linguagens Formais e Autômatos . 6. ed. Porto Alegre: Bookman, 2010. 256 p.

Título	Programação Paralela e Distribuída
Objetivo Geral	Familiarizar o estudante com os conceitos e termos básicos de sistemas paralelos, implementação e uso de concorrência, apresentar os tipos de arquitetura mais usados, descrever o suporte necessário para a programação de tais sistemas e apresentar algumas aplicações.
Pré-requisitos	Sistemas Operacionais
Disc. recomendadas	Sistemas Distribuídos
Carga horária total	60h
Natureza	▷ Teóricas: 30h ▷ Práticas: 30h
Caráter	Obrigatório
Ementa	Revisão de arquiteturas paralelas: memória compartilhada e distribuída. Desenvolvimento de aplicações concorrentes: conceitos básicos da programação concorrente, definição, ativação e coordenação de processos, modelos de programação e técnicas de decomposição. Técnicas de otimização. Otimização sequencial: uso eficiente da memória, unit stride, blocking. Instruções vetoriais e super escalares, opções de otimização. Profiling e modelagem de desempenho. Controle de processos e paralelização fork-join. Programação com memória compartilhada e introdução ao OpenMP. Programação com memória distribuída e MPI. Programação de sistemas <i>manycore</i> como GPU e aceleradores: CUDA, OpenCL e outros. Programação paralela na nuvem. Avaliação de desempenho e teste de programas concorrentes.
Respons. pela oferta	DC (1001483)
Bibliografia Básica	<p>GRAMA, A.; GUPTA, A.; KARYPIS, G.; KUMAR, V. <i>Introduction to Parallel Computing</i>. Boston: Addison-Wesley, 2003.</p> <p>DONGARRA, J.; FOSTER, I.; FOX, G.; GROPP, W.; WHITE, A.; TORCZON, L.; KENNEDY, K. <i>Sourcebook of Parallel Computing</i>. San Francisco: Morgan Kaufmann, 2003.</p> <p>FOSTER, I. <i>Designing and Building Parallel Programs</i>. Boston: Addison-Wesley, 1995. Disponível em: http://www-unix.mcs.anl.gov/dbpp.</p> <p>CASANOVA, H.; LEGRAND, A.; ROBERT, Y. <i>Parallel Algorithms</i>. Boca Raton: CRC Press, 2009. 335 p.</p> <p>WILKINSON, B.; ALLEN, M. <i>Parallel Programming: Techniques and Applications Using Networked Workstations and Parallel Computers</i>. Boston: Pearson Prentice Hall, 2005.</p> <p>QUINN, M. J. <i>Parallel Programming: in C with MPI and OpenMP</i>. Boston: McGraw-Hill Higher Education, 2004.</p> <p>LIN, C.; SNYDER, L. <i>Principles of Parallel Programming</i>. Boston: Pearson Addison Wesley, 2009.</p>

Bibliografia Complementar

- FLYNN, M. J.; RUDD, K. W. *Parallel Architectures*. ACM Computing Surveys, v. 28, n. 1, 1996.
- CHAPMAN, B.; JOST, G.; VAN DER PAS, R. *Using OpenMP: Portable Shared Memory Parallel Programming*. Cambridge: MIT Press, 2007.
- ROBBINS, K. A.; ROBBINS, S. *Practical Unix Programming: A Guide to Concurrency, Communication, and Multithreading*. Upper Saddle River: Prentice-Hall, 1996.
- STEVENS, W. R. *UNIX Network Programming: Interprocess Communications*. 2. ed. Upper Saddle River: Prentice Hall, 1999.
- STEVENS, W. R. *Unix Network Programming: Networking APIs: Sockets and XTI*. 2. ed. Upper Saddle River: Prentice Hall, 1999.
- SNIR, M. et al. *MPI - The Complete Reference: The MPI Core*. 2. ed. Cambridge: MIT Press, 1998.
- GROPP, W. et al. *MPI - The Complete Reference: The MPI Extensions*. 2. ed. Cambridge: MIT Press, 1998
-

Título	Paradigmas de Linguagens de Programação
Objetivo Geral	Familiarizar o estudante com diferentes paradigmas de programação, com foco na programação lógica e funcional. Habilitar o estudante a reconhecer as características, vantagens, desvantagens e aplicabilidade de cada paradigma em diferentes situações. Capacitar o estudante a desenvolver programas utilizando os paradigmas de programação lógica e funcional.
Pré-requisitos	Projeto e Análise de Algoritmos
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Teóricas: 30h ▷ Práticas: 30h
Caráter	Optativo
Ementa	Motivações para o estudo dos diferentes paradigmas de programação, a evolução histórica e alguns dos principais fatores que definem características das linguagens. Influências da arquitetura de máquina, das metodologias de desenvolvimento de software, e como se verificam as características de modularidade, extensibilidade, efeito colateral e o método de implementação da linguagem em cada paradigma. Conceitos de linguagens imperativas, como amarração de variáveis a escopo, tipo, memória e valor, métodos de passagem de parâmetros, aspectos de implementação de subprogramas, funcionamento da pilha de execução. Conceitos das linguagens de programação lógica, com foco nos aspectos de linguagem declarativa, interpretada, simbólica, e o uso das estruturas de listas e de recursão. Conceitos das linguagens de programação funcional, com foco nos aspectos de linguagem declarativa, interpretada, simbólica, e o uso das estruturas de listas e de recursão. Desenvolvimento de programas com versões imperativas, lógicas e funcionais.
Respons. pela oferta	DC (1001511)
Bibliografia Básica	<p>SEBESTA, R. W. Conceitos de linguagens de programação. [<i>Concepts of programming languages</i>]. José Carlos Barbosa dos Santos (Trad.). 5 ed. Porto Alegre: Bookman, 2003. 638 p. ISBN 85-363- 0171-6.</p> <p>GHEZZI, C.; JAZAYERI, M. Conceitos de linguagens de programação. [<i>Programming language concepts</i>]. Paulo A.S. Veloso (Trad.). Rio de Janeiro: Campus, 1985. 306 p. ISBN 85-7001- 204-7.</p> <p>SETHI, R. Programming languages: concepts and constructs. 2 ed. Reading: Addison-Wesley, 1996. 640 p. ISBN 0-201- 59065-4.</p> <p>Nicoletti, M. do C. A Cartilha da Lógica. 3. ed. LTC. 2017. 235p. ISBN 85-2162-693-2</p>
Bibliografia Complementar	<p>FURTADO, A. L. Paradigmas de linguagens de programação. Campinas: UNICAMP, 1986. 146 p.</p> <p>SILVA, J. C. G. da; ASSIS, F. S. G. de. Linguagens de programação: conceitos e avaliação; Fortran, C, Pascal, Modula-2, Ada, Chill. São Paulo: McGraw-Hill do Brasil, 1988. 213 p.</p> <p>BRATKO, I. Prolog: programming for artificial intelligence. 2 ed. Harlow: Addison-Wesley, 1990. 597 p. (<i>International Computer Science Series</i>) ISBN 0-201- 41606-9.</p> <p>WINSTON, P. H.; HORN, B. K. P. LISP. 2 ed. Reading: Addison-Wesley, 1984. 434 p. ISBN 0-201- 08372-8.</p>

Título	Desenvolvimento de Software para Web 2
Objetivo Geral	Familiarizar o estudante com os principais conceitos do desenvolvimento de software para web; capacitar o estudante a desenvolver aplicações web pelo lado do cliente (front-end).
Pré-requisitos	Desenvolvimento de Software para Web 1
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Práticas: 60h
Caráter	Optativo
Ementa	Criação de conteúdo Web. Formatação de conteúdo. Web responsiva. Programação Front-End. Automatização e gerenciamento de scripts de construção e implantação de aplicações front-end. Gerenciamento automático de pacotes e dependências em aplicações front-end. Frameworks para desenvolvimento Front-End.
Respons. pela oferta	DC (1001543)
Bibliografia Básica	
LAWSON, B. Introdução ao HTML 5 . Rio de Janeiro: Alta Books, 2011.	
SILVA, M. S. CSS3 : desenvolva aplicações web profissionais com uso dos poderosos recursos de estilização das CSS3. São Paulo: Novatec, 2012.	
SILVA, M. S. HTML 5 : a linguagem de marcação que revolucionou a web. São Paulo: Novatec, 2011.	
TERUEL, E. C. HTML 5 . São Paulo: Erica, 2012.	
PINHO, D. M. ECMA. Script 6 : Entre de cabeça no futuro do JavaScript. Casa do Código, 2017. ISBN: 978-85-5519-258-6. 227p.	
PONTES, G. Progressive Web Apps : Construa aplicações progressivas com React. Casa do Código, 2018. ISBN: 978-85-94188-54-0. 443p.	
Bibliografia Complementar	
https://www.w3schools.com/	

Título	Arquitetura de Software e Padrões
Objetivo Geral	Habilitar o aluno a identificar situações típicas para aplicação de padrões; habilitar o estudante a projetar e implementar padrões de projeto, inclusive seu uso combinado; habilitar o estudante a projetar a arquitetura de um sistema de software de forma a atender determinados atributos de qualidade; habilitar o estudante a identificar smells de código e identificar refatorações que possam corrigi-los.
Pré-requisitos	Engenharia de Software 2
Disc. recomendadas	—
Carga horária total	60h
Natureza	► Teóricas: 60h
Caráter	Optativo
Ementa	Apresentação de smells de código e refatorações. Introdução aos padrões de software. Aprofundamento em padrões GRASP e padrões de projeto: conceitos, implementação e combinação de padrões. Identificação de oportunidades para aplicação de padrões. Arquitetura de Software: conceituação e definições, abstrações, estilos arquiteturais, padrões arquiteturais; arquitetura de software e sua relação com atributos de qualidade (desempenho, manutenibilidade, disponibilidade, escalabilidade, etc).
Respons. pela oferta	DC (1001526)
Bibliografia Básica	
GAMMA, E.; HELM, R.; JOHNSON, R.; VLISSIDES, J. <i>Design patterns: elements of reusable object-oriented software</i> . Boston: Addison-Wesley, 2013, 395. ISBN 9780201633610.	
BASS, L.; CLEMENTS, P.; KAZMANN, R. <i>Software Architecture in Practice</i> . Third Edition. Addison-Wesley. 2013. Livro disponível gratuitamente em: https://smtebooks.com/file/8479	
FOWLER, Martin. <i>Refactoring: improving the design of existing code</i> . Addison-Wesley Professional, 2018.	
Bibliografia Complementar	
LIPPERT, M.; ROOCK, S. <i>Refactoring in large software projects</i> . Chichester: John Wiley & Sons, 2006. 280 p. ISBN 9780470858929.	
Richards. M. <i>Software Architecture Patterns</i> . O'Reilly, 2015. Livro disponível gratuitamente em http://www.oreilly.com/programming/free/files/software-architecture-patterns.pdf	
GARLAN, D.; SHAW, M. <i>An Introduction to Software Architecture</i> . CMU Software Engineering Institute Technical Report CMU/SEI-94-TR-21, ESC-TR-94-21, 1994, 39 p.	

Título	Banco de Dados para Ciência de Dados
Objetivo Geral	Capacitar o estudante com aprofundamento em conhecimentos em Banco de Dados para aplicá-lo em diversas fases do processo de análise de dados associado à Ciência de Dados. Familiarizar o estudante com os conceitos básicos de Big Data, banco de dados na nuvem, banco de dados NoSQL e outras alternativas ao modelo relacional; emprego de banco de dados explorando processamento paralelo e distribuído em clusters de computadores.
Pré-requisitos	Projeto e Implementação de Banco de Dados
Disc. recomendadas	—
Carga horária total	60h
Natureza	► Teóricas: 30h ► Práticas: 30h
Caráter	Optativo
Ementa	Introdução ao Big Data. Visão sobre o desenvolvimento de aplicações de banco de dados à nuvem. Explicitação sobre os modelos NoSQL: chave-valor, orientados a documentos, família de colunas e orientados a grafos. SGBDs NoSQL. Apresentação sobre banco de dados em um ambiente com processamento paralelo e distribuído em clusters de computadores.
Respons. pela oferta	DC (1001512)
Bibliografia Básica	
AMARAL, F. <i>Introdução à Ciência de Dados: Mineração de Dados e Big Data</i> . 1. ed. ISBN:8-57608-934-3, 2016.	
HARISSON, GUY. <i>Next Generation Databases: NoSQL, NewSQL and Big Data</i> . Apress. 2015. ISBN: 978-1-4842-1329-2.	
REDMOND, E.; WILSON, J. R. <i>Seven Databases in Seven Weeks: A Guide to Modern Databases and the NoSQL Movement</i> . 1. ed. Pragmatic Bookshelf, 2012. ISBN-10: 1934356921.	
Bibliografia Complementar	
SADALAGE, P. J.; FOWLER, M. <i>NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistent</i> . Addison-Wesley. 2013. ISBN: 978-0-321-82662-6.	
LEMAHIEU W.; VANDEN BROUCKE S.; BAESENS B. <i>Principles of Database Management: The Practical Guide to Storing, Managing and Analyzing Big and Small Data</i> . Cambridge University Press, 2018. ISBN 1107186129	
CHEN, Y.; KU, W.; FENG, J.; LIU, P.; SU, Z. <i>Secure Distributed Data Storage in Cloud Computing</i> . BUYYA, R.; BROBERG, J.; GOSCINSKI, A. (eds). Cloud Computing. New York: Wiley, 2011. doi:10.1002/9780470940105.ch8.	
RATNER, B. <i>Statistical modeling and analysis for database marketing: effective techniques for mining big data</i> . Boca Raton, Fla.: Chapman & Hall, c2003. 362 p. ISBN 1-57444-344-5.	
<i>ANALYSIS of symbolic data: exploratory methods for extracting statistical information from complex data</i> . New York: Springer, 2000. 425 p. (Studies in Classification, Data Analysis, and Knowledge Organization). ISBN 3-540-66619-2.	

Título	Aprendizado de Máquina 2
Objetivo Geral	Familiarizar o estudante com os conceitos e algoritmos avançados de aprendizado de máquina supervisionado e não-supervisionado; proporcionar ao estudante o aprofundamento em paradigmas e problemas complexos do aprendizado de máquina.
Pré-requisitos	Aprendizado de Máquina 1
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Teóricas: 60h
Caráter	Optativo
Ementa	Apresentação de problemas reais tratados com Aprendizado de Máquina. Noções gerais sobre combinação de classificadores. Introdução às redes neurais artificiais. Aprofundamento em algoritmos de classificação. Introdução ao aprendizado em fluxo de dados. Apresentação de conceitos e algoritmos de computação evolutiva aplicada ao Aprendizado de Máquina. Noções gerais sobre técnicas de aprendizado ativo. Visão geral sobre outras abordagens de aprendizado de máquina.
Respons. pela oferta	DC (1001513)
Bibliografia Básica	
DENG, L; YU, D. <i>Deep learning: methods and applications. Foundations and Trends® in Signal Processing.</i> v. 7, n. 3–4, p. 197-387, 2014.	
GAMA, J. <i>Knowledge discovery from data streams</i> . CRC Press, 2010.	
FREITAS, A. A. <i>Data mining and knowledge discovery with evolutionary algorithms</i> . Springer Science & Business Media, 2013.	
Bibliografia Complementar	
DE CASTRO, L. N. <i>Fundamentals of natural computing: basic concepts, algorithms, and applications</i> . CRC Press, 2006.	
ZHANG, C.; MA, Y. (Ed.). <i>Ensemble machine learning: methods and applications</i> . Springer Science & Business Media, 2012.	
EIBEN, A. E. et al. <i>Introduction to evolutionary computing</i> . Heidelberg: Springer, 2003.	

Título	Processamento e Visualização de Dados
Objetivo Geral	Capacitar o estudante a entender e trabalhar com os procedimentos necessários para transformar dados possibilitando a análise e visualização destes por ferramentas computacionais, garantindo qualidade e minimizando distorções. Familiarizar o estudante com os princípios e técnicas de visualização da informação e como trabalhar com eficiência considerando os recursos gráficos atuais, por software e/ou hardware especializados.
Pré-requisitos	—
Disc. recomendadas	—
Carga horária total	60h
Natureza	► Teóricas: 30h ► Práticas: 30h
Caráter	Optativo
Ementa	Introdução ao conceito de conjuntos de dados e aprofundamento na análise estatística apropriada para técnicas de mineração de dados. Apresentação dos modelos básicos de preparação de dados. Apresentação de técnicas para lidar com valores ausentes e com dados ruidosos. Técnicas para redução de dados; seleção de atributos e instâncias. Amostragem. Discretização. Introdução a dados tabulares. Modelos de projeções multidimensionais, hierárquicas e gráficos tridimensionais.
Respons. pela oferta	DC (1001514)
Bibliografia Básica	<p>WARE, C. <i>Information Visualization</i>. 3. ed. Amsterdam: Elsevier, 2012.</p> <p>HAN, J.; KAMBER, M.; PEI, J. <i>Data Mining: Concepts and Techniques</i>. 3. ed. San Francisco: Morgan Kaufmann, 2011.</p> <p>GARCÍA, S.; LUENGO, J.; HERRERA, F. <i>Data Preprocessing in Data Mining</i>. (<i>Intelligent Systems Reference Library</i>). Berlin: Springer, 2015.</p>
Bibliografia Complementar	<p>SPENCE, R. <i>Information Visualization: Design for Interaction</i>. 2. ed. Upper Saddle River: Prentice Hall, 2007.</p> <p>VANDERPLAS, J. <i>Python Data Science Handbook: Essential Tools for Working with Data</i>. Sebastopol: O'Reilly Media, 2016.</p> <p>TELEA, A. <i>Data Visualization: Principles and Practice</i>. Natick: A.K. Peters, 2007.</p> <p>WARD, M. O.; GRINSTEIN, G.; KELM, D. <i>Interactive Data Visualization</i>. Natick: A.K. Peters, 2010.</p> <p>GERALD, F.; HANSFORD, D. <i>Mathematical Principles for Scientific Computing and Visualization</i>. Natick: A.K. Peters, 2008.</p> <p>DE JONGE, E.; VAN DER LOO, M. P. J. <i>An Introduction to Data Cleaning with R</i>. 2013. Disponível em: https://cran.r-project.org/doc/contrib/de_Jonge+van_der_Loo-Introduction_to_data_cleaning_with_R.pdf. Acesso em: 06 Jun. 2024.</p> <p><i>7 Steps to Mastering Data Preparation with Python</i>. Disponível em: https://www.kdnuggets.com/2017/06/7-steps-mastering-data-preparation-python.html. Acesso em: 06 Jun. 2024.</p> <p>LEVINE, D. C. et al. <i>Estatística: Teoria e Aplicações</i>. 5. ed. Rio de Janeiro: LTC, 2008.</p>

Título	Processamento Digital de Imagens 3D e Vídeos
Objetivo Geral	Habilitar o estudante apto a visualizar e processar imagens tridimensionais, bem como sequências temporais de imagens (vídeos); habilitar o estudante a aplicar técnicas eficientes de processamento de imagens, essenciais para a análise de imagens 3D e vídeos.
Pré-requisitos	Processamento Digital de Imagens
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Teóricas: 30h ▷ Práticas: 30h
Caráter	Optativo
Ementa	Revisão sobre técnicas básicas de processamento de imagens. Apresentação de ferramentas e técnicas de visualização de imagens 3D e vídeo. Visão geral sobre formatos de vídeos. Aprofundamento sobre os desafios encontrados em imagens 3D não-isotrópicas. Análise de movimento (estimação e estabilização de movimento, fluxo ótico, rastreamento de objetos). Processamento espaço-temporal. Apresentação sobre técnicas de interpolação. Detalhamento sobre cálculo de esqueleto em 3D. Visão geral de técnicas de detecção de estruturas tubulares.
Respons. pela oferta	DC (1001480)
Bibliografia Básica	
FORSYTH, D. A.; PONCE, J. <i>Computer Vision: A Modern Approach</i> . Upper Saddle River: Prentice Hall, 2003.	
GONZALEZ, R. C.; WOODS, R. E. <i>Digital Image Processing</i> . 3. ed. Upper Saddle River: Prentice-Hall, 2008.	
BOVIK, A. <i>Handbook of Image and Video Processing</i> . 2. ed. Amsterdam: Elsevier Academic Press, 2005.	
Bibliografia Complementar	
KAEBLER, A.; BRADSKI, G. <i>Learning OpenCV - Computer Vision in C++ with the OpenCV Library</i> . 1. ed. Sebastopol: O'Reilly, 2017.	
WOODS, J. W. <i>Multidimensional Signal, Image, and Video Processing and Coding</i> . Amsterdam: Elsevier, 2006.	
TEKALP, A. M. <i>Digital Video Processing</i> . Upper Saddle River: Prentice Hall Press, 2015.	
OPENHEIM, A. V.; SCHAFER, R. W. <i>Discrete-Time Signal Processing</i> . Upper Saddle River: Prentice-Hall, 1989.	
PROAKIS, J. G.; MANOLAKIS, D. G. <i>Digital Signal Processing: Principles, Algorithms and Applications</i> . New York: MacMillan, 1992.	

Título	Arquitetura e Organização de Computadores 2
Objetivo Geral	Ao final da disciplina o estudante deve ser capaz de entender a organização das principais arquiteturas modernas, bem como as técnicas de extração de paralelismo para o desenvolvimento visando alto desempenho.
Pré-requisitos	Arquitetura e Organização de Computadores 1
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Teóricas: 30h ▷ Práticas: 30h
Caráter	Optativo
Ementa	Linguagem de máquina de processadores modernos; Níveis de paralelismo: ILP, execução fora de ordem, SIMD, thread. Programação de baixo nível (<i>System Programming</i>) e Suporte ao Sistema Operacional. Interfaces de E/S, interrupções e timers.
Respons. pela oferta	DC (1001541)
Bibliografia Básica	
HENNESSY, J. L.; PATTERSON, David A. Arquitetura de computadores: uma abordagem quantitativa . 3. ed. Rio de Janeiro: Campus, 2003. 827 p. ISBN 85-352-1110-1.	
STALLINGS, W. Arquitetura e organização de computadores : projeto para o desempenho. 5. ed. São Paulo: Prentice Hall, 2002. 786 p. ISBN 85-87918-53-2.	
HYDE, R. The art of assembly language . San Frascisco: No Starch Press, c2003. 903 p. ISBN 1-886411-97-2.	
IRVINE, Kip R. Assembly language for intel-based computers . 5. ed. Upper Saddle River: Prentice Hall, c2007. 722 p. ISBN 0-13-238310-1.	
Bibliografia Complementar	
PATTERSON, D. A.; HENNESSY, J. L. Organização e projeto de computadores : a interface harware/software. 3. ed. Rio de Janeiro: Elsevier, 2005. 484 p. ISBN 8535215212.	
HARRIS, D. M.; HARRIS, S. L. Digital design and computer architecture . San Frascisco: Elsevier, 2007. 569 p. ISBN 978-0-12-370497-9.	
STALLINGS, W. Arquitetura e organização de computadores . 8. ed. São Paulo: Pearson, 2012. 624 p. ISBN 978-85-7605-564-8.	

Título	Sistemas Digitais
Objetivo Geral	Ao final da disciplina o estudante deve ser capaz de projetar, analisar e testar circuitos digitais síncronos e assíncronos complexos, e aplicar técnicas para solução de problemas inerentes a sua implementação.
Pré-requisitos	Lógica Digital
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Teóricas: 30h ▷ Práticas: 30h
Caráter	Optativo
Ementa	
Máquinas de estado algorítmicas. Arquiteturas eficientes dedicadas para solução de problemas (não von Neumann). Sincronização de relógio e temporização. Instabilidades e Falhas (Hazards e Glitches). Tecnologia de Implementação. Avaliação e testes de circuitos: Placas de Circuito Impresso, Ruídos, Alta frequência, Built-in Self-Test, Transmission-Line Effects.	
Respons. pela oferta	DC (1001539)
Bibliografia Básica	
PEDRONI, V. A. Eletrônica digital moderna e VHDL : princípios digitais, eletrônica digital, projeto digital, microeletrônica e VHDL. Rio de Janeiro: Elsevier, 2010. 619 p. ISBN 9788535234657.	
ARNOLD, M. G. Verilog digital computer design: Algorithms into hardware . Upper Saddle River: Prentice Hall PTR, c1999. 602 p. ISBN 0-13-639253-9.	
HAMBLEN, J. O.; HALL, T. S.; FURMAN, M. D. Rapid prototyping of digital systems . New York: Springer, 2008. xvii, 411 p. : il., graf., ISBN 978038772670.	
Bibliografia Complementar	
BROWN, S. D. Fundamentals of digital logic with Verilog design . Tata McGraw-Hill Education, 2007. LALA, P. K. Self-checking and Fault-tolerant Digital Design . San Francisco: Morgan Kaufmann, 2001.	
HARRIS, D. Skew-tolerant Circuit Design . San Francisco: Morgan Kaufmann, 2001.	
OKLOBDZIJA, V. G., ed. Digital Design and Fabrication . Boca Raton: CRC Press, 2017.	
AGARWAL, A.; LANG, J. H. Foundations of Analog and Digital Electronic Circuits . Amsterdam: Elsevier, 2005.	
WOLF, M. Computers as Components: Principles of Embedded Computing System Design . 2. ed. Amsterdam: Elsevier, 2012.	
SZALAPAJ, P. Contemporary Architecture and the Digital Design Process . New York: Routledge, 2014.	

Título	Segurança Cibernética
Objetivo Geral	Gerar capacitação para entender, analisar e projetar técnicas de exploração de falhas de segurança de sistemas cibernéticos. Gerar competências para abordagem e proteção de sistemas computacionais, utilizando técnicas de exploração mais comuns. Capacitar para projeto e análise de sistemas computacionais seguros.
Pré-requisitos	Sistemas Operacionais e Arquitetura e Organização de Computadores 1
Disc. recomendadas	—
Carga horária total	60h
Natureza	► Teóricas: 30h ► Práticas: 30h
Caráter	Optativo
Ementa	Introdução à segurança de sistemas. Arquiteturas para segurança: segurança para aplicativos, sistemas operacionais e códigos legados, isolamento, controle de acesso. Criptografia: encriptação, identificação, autenticação, integridade, não repudiação, infraestrutura de chaves públicas (PKI). Segurança web: modelo de segurança de serviços e de navegadores web; Vulnerabilidades comuns (Top OWASP), tais como: SQL Injection, XSS e CSRF. Segurança de software: compilação e semântica de execução, ataques de controle de fluxo, defesas contra ataques de controle de fluxo, ROP, integridade de controle de fluxo (CFI). Segurança de rede: monitoramento, detecção de intrusão (IDS) e arquitetura de redes seguras. Tópicos avançados, tais como: segurança de aplicações móveis, módulos SAM, SIM, JavaCard e Contactless Smart Cards, e-Wallets, EMV e sistemas de bilhetagem eletrônica.
Respons. pela oferta	DC (1001520)
Bibliografia Básica	<p>STALLINGS, W. <i>Criptografia e segurança de redes: princípios e práticas</i>. 4. ed. São Paulo: Pearson, 2010. ISBN 9788576051190.</p> <p>ERICKSON, J.M. <i>Hacking: the art of exploration</i>. San Frascisco: No Starch Press, 2003. ISBN 1-59327-007-0.</p> <p>NICHOLS, Randall K. <i>ICSA guide to cryptography</i>. New York: McGraw-Hill, 1999. ISBN 0-07-913759-8.</p> <p>NORTHCUTT, S.; NOVAK, J.; MCLACHLAN, D. <i>Network intrusion detection: an analyst's handbook</i>. 2. ed. Indianapolis: New Riders, 2000. ISBN 0-7357-1008-2.</p> <p>STAJANO, F. <i>Security for ubiquitous computing</i>. Chichester: John Wiley & Sons, c2002. ISBN 0-470-84493-0.</p>
Bibliografia Complementar	<p>MEAD, N. R. & WOODY, C. C. <i>Cyber Security Engineering – A Practical Approach for Systems and Software Assurance</i>. Addison-Wesley, 2017.</p> <p>SCHNEIER, B. <i>Applied Cryptography, Protocols, Algorithms and Source Code in C</i>. Published by John Wiley and Sons, 1996 – Reprinted in 2016.</p> <p>KATZ, J. <i>Introduction to Modern Cryptography</i>. CRC Press 2015.</p> <p>WYKES, S. M. <i>Criptografia Essencial, A Jornada do Criptógrafo</i>. Elsevier, 2016.</p> <p>STALLINGS, W. <i>Criptografia e Segurança de Redes</i>. Editora Pearson, 2007.</p> <p>NEMADI, H. R.; YANG, L. <i>Applied Cryptography for Cyber Security and Defense: Information Encryption and Cyphering</i>. Premier Reference Source, 2010.</p>

Título	Otimização Matemática
Objetivo Geral	Desenvolver competências nos seguintes tópicos da área de otimização: modelagem e análise e resolução de problemas de otimização lineares e não lineares; gerar capacitação para resolução de tais problemas de forma analítica e computacional. Abordagem a partir de versões aproximadas das estratégias exatas.
Pré-requisitos	Cálculo 2
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Teóricas: 60h
Caráter	Optativo
Ementa	Programação Linear: Método simplex, Dual do Problema, Dualidade. Programação Inteira. Método <i>Branch-and-Bound</i> . Programação não linear: com e sem restrições. Método gradiente conjugado e Hessiano. Multiplicadores de Lagrange, Fluxo em Redes: algoritmos Ford-Fulkerson e Edmonds-Karp (teorema min cut/max flow); Emparelhamento máximo em grafos bipartidos: algoritmo húngaro. Métodos dos mínimos quadrados e regressão linear. Teoria das filas. Simulação de Eventos Discretos.
Respons. pela oferta	DC (1001346)
Bibliografia Básica	
KLEINBERG, J.; TARDOS, E. Algorithm Design . Pearson New International Edition. Pearson Education Limited, 2013. 832 p. ISBN 9781292037042.	
CORMEN, T.H.; LEISERSON, C.E.; RIVEST, R.L.; STEIN, C. Introduction to Algorithms , 3. ed. McGraw-Hill, 2009.	
LACHTERMACHER, G. Pesquisa Operacional na Tomada de Decisões . 4. ed. São Paulo: Pearson Prentice Hall, 2013. 223 p. ISBN 9788576050933.	
ARENALES, M.N.; ARMENTANO, V.A.; MORABITO, R.; YANASSE, H. Pesquisa Operacional . Rio de Janeiro: Elsevier, 2007. 524 p. (Coleção CAMPUS-ABREPO Engenharia de Produção). ISBN 85-352-1454-3.	
TAHA, H.A. Pesquisa Operacional . 8. ed. São Paulo: Pearson, 2008. xiii, 359 p. ISBN 9788576051503.	
Bibliografia Complementar	
HILLIER, F.S.; LIEBERMAN, G.J. Introdução à Pesquisa Operacional . 8. ed. São Paulo: McGraw Hill, 2006. 828 p. ISBN 8586804681.	
ANDRADE, E.L. Introdução à Pesquisa Operacional: Métodos e Modelos para Análise de Decisões . 3. ed. Rio de Janeiro: LTC, 2004. xiii, 192 p. ISBN 8521614128.	
ELLENRIEDER, A.V. Pesquisa Operacional . Rio de Janeiro: Almeida Neves-Editores, 1971. 261 p.	

4.5.7 Disciplinas do Sétimo Semestre

Título	Construção de Compiladores
Objetivo Geral	Habilitar o estudante a não ser apenas um utilizador de linguagens existentes, mas sim um projetista; capacitar o estudante com habilidades para criar suas próprias linguagens para situações de diferentes domínios. Desenvolver no estudante a competência para construir um compilador completo utilizando ferramentas de auxílio à construção automática.
Pré-requisitos	Teoria da Computação e Construção de Algoritmos e Programação
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Teóricas: 60h
Caráter	Obrigatório
Ementa	Apresentação e contextualização sobre Compiladores. Visão geral sobre a estrutura de um compilador (etapas de front-end/análise e etapas de back-end/síntese). Detalhamento da etapa de Análise Léxica. Detalhamento da etapa de Análise Sintática: Análise Sintática Descendente. Análise Sintática Ascendente. Detalhamento da etapa de Análise semântica. Detalhamento da etapa de Geração e otimização de código. Noções de Manipulação de erros. Apresentação de algumas ferramentas de auxílio à construção de um compilador. Aprofundamento no projeto e na implementação de um compilador completo, traduzindo uma linguagem de programação simplificada para código executável em arquitetura física ou virtual. Aprofundamento no projeto e na implementação de um compilador (análise léxica, análise sintática, análise semântica e geração de código ou interpretação) para um domínio de escolha do estudante.
Respons. pela oferta	DC (1001497)
Bibliografia Básica	
ALFRED V. AHO. et al. Compiladores: princípios, técnicas e ferramentas . 2. ed. São Paulo: Pearson Addison-Wesley, 2007. x, 634 ISBN 9788588639249 -	
LOUDEN, K. C. Compiladores: princípios e práticas . São Paulo: Pioneira Thomson Learning, 2004. 569 p. ISBN 85-221-0422-0.	
COOPER, K. D.; Torczon, L. Engineering a compiler . 2. ed. Amsterdam: Elsevier, 2012. xxiii, 800 p. ISBN 9780120884780.	
Bibliografia Complementar	
NETO, J. J. Introdução à Compilação . 2. ed. Rio de Janeiro: Elsevier, 2016. 307 p. ISBN 9788535278101.	
PARR, T. The Definitive ANTLR 4 Reference . IN: The Pragmatic Bookshelf . 2013. 328 p. ISBN 9781934356999.	
DELAMARO, M.E. Como Construir um Compilador Utilizando Ferramentas Java . IN: Novatec, 2004. 308p. ISBN 8575220551.	
MAK, R. Writing compilers and interpreters: a modern software engineering approach using Java . 3. ed. Indianapolis, IN: Wiley Publishing, 2009. xxiii, 840 p. ISBN 9780470177075.	

Título	Desenvolvimento Móvel
Objetivo Geral	Familiarizar o estudante com os conceitos da programação para dispositivos móveis; familiarizar o estudante com conceitos de programação multiplataforma; capacitar o estudante a desenvolver aplicativos para dispositivos móveis.
Pré-requisitos	Desenvolvimento de Software para Web 2
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Práticas: 60h
Caráter	Optativo
Ementa	
Características e evolução dos dispositivos móveis. Versionamento em aplicações móveis. Modelos arquiteturais da programação móvel. Programação da interface para dispositivos móveis. Comunicação e sincronização de dados. Persistência de dados no dispositivo. Utilização dos recursos de hardware do dispositivo móvel. Frameworks de programação para dispositivos móveis.	
Respons. pela oferta	DC (1001337)
Bibliografia Básica	
LAWSON, B. Introdução ao HTML 5 . Rio de Janeiro: Alta Books, 2011.	
LEE, V.; SCHENEIDER, H.; SCHELL, R. Aplicações móveis: arquitetura, projeto e desenvolvimento . São Paulo: Pearson Education: Makron Books, 2015. 328 p.	
SILVA, M. S. CSS3 : desenvolva aplicações web profissionais com uso dos poderosos recursos de estilização das CSS3. São Paulo: Novatec, 2012.	
SILVA, M. S. HTML 5 : a linguagem de marcação que revolucionou a web. São Paulo: Novatec, 2011.	
SILVA, M. S. J. Query Mobile : desenvolva aplicações web para dispositivos móveis com HTML5, CCSS3, AJAX, jQuery e jQuery UI. São Paulo: Novatec, 2012.	
TERUEL, E. C. HTML 5 . São Paulo: Erica, 2012.	
Bibliografia Complementar	
BORGES JÚNIOR, M. P. Aplicativos móveis: aplicativos para dispositivos móveis usando C#.Net com a ferramenta visual Studio.NET e MySQL e SQL Server. Rio de Janeiro: Ciência Moderna, 2005. 130p.	
DEITEL, H. M.; DEITEL, P. J. Java: como programar . 8. ed. São Paulo: Bookman, 2010.	
FLATSCHART, F. HTML 5: embarque imediato . Rio de Janeiro: Brasport, 2011.	
LECHETA, R. R. Google Android: aprenda a criar aplicações para dispositivos móveis com o Android SDK . 3. ed. São Paulo: Novatec, 2013.	

Título	DevOps
Objetivo Geral	Habilitar o estudante a utilizar ferramentas e conhecimentos necessários para automatizar processos e infraestruturas de engenharia de software. Capacitar o estudante a integrar ferramentas existentes ou desenvolver novas ferramentas de engenharia de software. Habilitar o estudante a projetar um pipeline DevOps. Capacitar o estudante a utilizar o pipeline desenvolvido para práticas de projetos de Engenharia de Software.
Pré-requisitos	Engenharia de Software 2
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Práticas: 60h
Caráter	Optativo
Ementa	Introdução aos conceitos de desenvolvimento e operações (DevOps). Aprofundar conceitos e práticas de Gerência de Configuração de Software. Aprofundar conceitos e práticas de Construção de Software. Aprofundar conceitos e práticas de Teste de Software. Aprofundar conceitos e práticas de Análise de Software por meio de métricas. Compreender o conceito de entrega contínua de software. Compreender o conceito de pipeline DevOps. Integrar ferramentas e práticas para viabilizar um pipeline DevOps.
Respons. pela oferta	DC (1001544)
Bibliografia Básica	
DAVIS, J.; DANIELS, R. <i>Effective DevOps: Building a Culture of Collaboration, Affinity, and Tooling at Scale</i> . O'Reilly Media, 2016.	
KIM, G.; DEBOIS, P.; WILLIS, J.; HUMBLE, J.. <i>The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations</i> . IT Revolution, 2016.	
BEYER, B.; JONES, C.; PETOFF, J.; MURPHY, N. R. Site <i>Reliability Engineering: How Google Runs Production Systems</i> . O'Reilly Media, 2016.	
Bibliografia Complementar	
DUVALL, P.; MATYAS, S. M.; GLOVER, A. <i>Continuous Integration: Improving Software Quality and Reducing Risk</i> . The Addison-Wesley Signature Series. 2007.	
BASS, L.; WEBER, I.; ZHU, L.. <i>DevOps: A Software Architect's Perspective</i> . Addison-Wesley, 2015.	
<i>Continuous Delivery</i> . Disponível em: https://continuousdelivery.com/ . Acesso em: 23/04/2018.	

Título	Processamento de Linguagem Natural
Objetivo Geral	Capacitar o estudante a implementar aplicações computacionais de processamento de textos escritos em linguagens humanas (linguagem natural). Propiciar ao estudante a aquisição de conhecimento a respeito dos principais conceitos, técnicas e ferramentas para o processamento de linguagem natural (PLN). Desenvolver no estudante a competência para identificar problemas que podem ser resolvidos com técnicas de PLN e quais técnicas são adequadas a cada problema. Habilitar o estudante a seguir preceitos éticos no desenvolvimento de aplicações de PLN.
Pré-requisitos	Inteligência Artificial
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Teóricas: 60h
Caráter	Optativo
Ementa	Apresentação de uma visão geral do PLN e suas aplicações, bem como dos conceitos e terminologia básicos, do histórico de evolução da área, dos desafios e das limitações ainda existentes. Noções de linguística e introdução ao pré-processamento automático de textos escritos em linguagem humana. Detalhamento de técnicas de representação textual baseadas em representação vetorial e em características extraídas do texto. Noções de tarefas e técnicas de PLN como reconhecimento de entidades nomeadas, análise de sentimentos, classificação textual, tradução automática, sistemas de perguntas e respostas e sumarização automática. Apresentação de técnicas de aprendizado de máquina aplicadas ao PLN, incluindo aprendizado supervisionado, não-supervisionado, neural e profundo. Noções gerais sobre aplicações de PLN em domínios específicos, questões éticas e direções futuras no PLN.
Respons. pela oferta	DC (1003315)
Bibliografia Básica	
JURAFSKY, D.; MARTIN, J. H. <i>Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition</i> . 3rd. ed. USA: Prentice Hall PTR, 2023.	
MANNING, C. D.; SCHÜTZE, H. <i>Foundations of statistical natural language processing</i> . Cambridge, USA: mitpress, 1999.	
FREITAS, C. <i>Linguística Computacional</i> . [s.l.] Parábola Editorial, 2022.	
CASELI, H.M.; NUNES, M.G.V. (org.) <i>Processamento de Linguagem Natural: Conceitos, Técnicas e Aplicações em Português</i> . 2 ed. BPLN, 2024. Disponível em: https://brasileiraspln.com/livro-pln/2a-edicao .	
FACELI, K.; LORENA A.C.; GAMA J.; CARVALHO A.C P.L.F. <i>Inteligência Artificial - Uma Abordagem de Aprendizado de Máquina</i> . LTC. 2011.	
Bibliografia Complementar	
FERREIRA, M.; LOPES, M. <i>Para conhecer: Linguística Computacional</i> . [s.l.] Editora Contexto, 2019.	
MITKOV, R. <i>The Oxford handbook of Computational Linguistics</i> . [s.l.] Oxford University Press, 2003.	

Título	Visão Computacional
Objetivo Geral	Habilitar o estudante para o uso de ferramentas exatas e heurísticas de inferência baseada em imagens. Capacitar o estudante a resolver problemas práticos como reconhecimento facial, criação de mosaicos de imagens, detecção de movimento e reconhecimento de objetos em imagens. Capacitar o estudante a identificar as soluções de Visão Computacional mais apropriadas para cada problema.
Pré-requisitos	Processamento de Sinais Digitais ou (Álgebra Linear e Processamento Digital de Imagens), Aprendizado de Máquina 2
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Teóricas: 30h ▷ Práticas: 30h
Caráter	Optativo
Ementa	Revisão sobre técnicas básicas de processamento de imagens. Detalhamento sobre detecção de pontos salientes em imagens e vídeos. Visão geral sobre casamento de pontos salientes. Apresentação de técnicas de rastreamento. Visão geral sobre detecção de movimento em vídeos. Apresentação de métodos de Aprendizado de Máquina para Visão Computacional aplicados em segmentação, identificação de objetos e classificação de imagens.
Respons. pela oferta	DC (1001532)
Bibliografia Básica	<p>FORSYTH, D. A.; PONCE, J. <i>Computer Vision: A Modern Approach</i>. Prentice Hall, 2003.</p> <p>GONZALEZ, R. C.; WOODS, R. E. <i>Digital Image Processing</i>. 3. ed., Prentice-Hall, 2008.</p> <p>BALLARD, D.; BROWN, C. <i>Computer Vision</i>. Prentice Hall, 1982.</p>
Bibliografia Complementar	<p>SZELISKI, R. <i>Computer Vision: Algorithms and Applications</i>. Springer, 2010. Disponível em: (http://szeliski.org/Book/). Acesso em: 17 Jun. 20024.</p> <p>KAEBLER, A.; BRADSKI, G. <i>Learning OpenCV - Computer Vision in C++ with the OpenCV library</i>. 1. ed., O'Reilly, 2017.</p> <p>MIXON, M.; AGUADO, A. S. <i>Feature Extraction & Image Processing for Computer Vision</i>. 2. ed., Academic Press, 2008.</p> <p>OPENHEIM, A. V.; SCHAFER, R. W., <i>Discrete-Time Signal Processing</i>. Prentice-Hall, 1989.</p> <p>PROAKIS, J. G.; MANOLAKIS, D. G. <i>Digital Signal Processing: Principles, Algorithms and Applications</i>. MacMillan, 1992.</p>

Título	Projeto de Sistemas Computacionais Embarcados
Objetivo Geral	Ao final da disciplina o estudante deve ser capaz de entender os conceitos, elementos, problemas e soluções típicas no desenvolvimento de sistemas computacionais embarcados. Entender o princípio de operação, configuração, vantagens e desvantagens dos periféricos mais utilizados em sistemas computacionais. Projetar, analisar e testar o hardware e o software de sistemas computacionais embarcados e de aplicar técnicas para solução de problemas inerentes a estes sistemas.
Pré-requisitos	Arquitetura e Organização de Computadores 2 e Engenharia de Sistemas
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Práticas: 60h
Caráter	Optativo
Ementa	Conceitos e aplicações de sistemas computacionais embarcados. Metodologias para o desenvolvimento de Sistemas Embarcados: engenharia dirigida por modelos, AADL, SysML. Co-projeto de hardware e software. Ciclo de desenvolvimento de software: diagramas de fluxo de dados, statecharts, redes de petri temporizadas. Sensores, conversores, atuadores e outros componentes típicos. Microkernels: multitarefa, escalonamento e sincronização. Sistemas Críticos: RTOS, tolerância a falhas, redundância, certificação. Geração automática de código. Testes e simulação: hardware e software in the loop. Exemplos práticos de projeto de sistemas embarcados. Prototipação.
Respons. pela oferta	DC (1001538)
Bibliografia Básica	<p>WOLF, W. <i>Computers as Components: Principles of Embedded Computing System Design</i>. San Francisco: Morgan Kaufmann, c2005. 6556 p. ISBN 0-12-369459-0.</p> <p>BALL, S.R. <i>Analog Interfacing to Embedded Microprocessor Systems</i>. 2n. ed. Boston: Newnes, c2004. 322 p. (Embedded Technology Series). ISBN 978-0-7506-7723-3.</p> <p>BRÄUNL, T. <i>Embedded Robotics: Mobile Robot Design and Applications with Embedded Systems</i>. 2. ed. Berlin: Springer-Verlag, c2006. 458 p. ISBN 3-540-34318-0.</p> <p>QING, L.; CAROLINE, Y. <i>Real-Time Concepts for Embedded Systems</i>. San Francisco: CMP Books, c2003. 294 p. ISBN 978-1-57820-124-2.</p> <p>HOLT, J.; PERRY, S. <i>SysML for Systems Engineering: A Model-Based Approach</i>. 2. ed. Stevenage: Institution of Engineering and Technology, 2013. 930 p. (Professional Applications of Computing Series; 7). ISBN 978-1-84919-651-2.</p>
Bibliografia Complementar	<p>MARWEDEL, P. <i>Embedded System Design: Embedded Systems Foundations of Cyber-Physical Systems</i>. and the Internet of Things, 3. ed. 2018 Edition.</p> <p>KORDON, F.; HUGUES, J.; CANALS, A.; DOHET, A. <i>Embedded Systems: Analysis and Modeling with SysML</i>. UML and AADL, 1. ed. 2013.</p> <p>VAHID, F.; GIVARGIS, T. <i>Embedded System Design: A Unified Hardware/Software Introduction</i>. Wiley, 2002.</p>

Título	Arquiteturas de Alto Desempenho
Objetivo Geral	Ao final da disciplina o estudante deve ser capaz de entender e projetar os principais tipos de arquiteturas não convencionais para alto desempenho e baixo consumo energético.
Pré-requisitos	Arquitetura e Organização de Computadores 1
Disc. recomendadas	—
Carga horária total	60h
Natureza	► Teóricas: 30h ► Práticas: 30h
Caráter	Optativo
Ementa	Arquiteturas Heterogêneas: Aceleradores; ASIPs (Application Specific Instruction-set Processors); GPUs (Graphics Processing Units); DSPs (Digital Signal Processors); SoCs (System on Chip).
Respons. pela oferta	DC (1001537)
Bibliografia Básica	<p>LASTOVETSKY, A. L.; DONGARRA, J. J. <i>High-performance heterogeneous computing</i>. Hoboken, N.J.: Wiley, 2009. 267 p. (Wiley Series on Parallel and Distributed Computing). ISBN 9780470040393.</p> <p>KASTNER, R. <i>Arithmetic optimization techniques for hardware and software design</i>. 1 online resource (viii, 187 ISBN 9780511712180 (ebook).</p> <p>HENNESSY, J. L.; PATTERSON, D. A. <i>Arquitetura de computadores</i>: uma abordagem quantitativa. 3. ed. Rio de Janeiro: Campus, 2003. 827 p. ISBN 85-352-1110-1.</p> <p>STALLINGS, W. <i>Arquitetura e organização de computadores</i>: projeto para o desempenho. 5. ed. São Paulo: Prentice Hall, 2002. 786 p. ISBN 85-87918-53-2.</p>
Bibliografia Complementar	<p>PATTERSON, D. A.; HENNESSY, J. L. <i>Organização e Projeto de computadores</i>: a interface hardware/software. 3. ed. Rio de Janeiro: Elsevier, 2005. 484 p. ISBN 8535215212.</p> <p>HARRIS, D. M.; HARRIS, S. L. <i>Digital design and computer architecture</i>. San Frascisco: Elsevier, 2007. 569 p. ISBN 978-0-12-370497-9.</p> <p>WOODS, R. WILEY INTERSCIENCE (ONLINE SERVICE). <i>FPGA-based implementation of signal processing systems</i>. Chichester, U.K.: John Wiley & Sons, 2008. ISBN 9780470713785.</p>

Título	Processamento de Dados em Escala
Objetivo Geral	Familiarizar o estudante com os desafios, técnicas e ferramentas de processar dados em larga escala; proporcionar ao estudante uma visão geral dos problemas complexos enfrentados ao se processar dados com severos requisitos em termos de volume, velocidade e variedade e sobre os paradigmas e ferramentas disponíveis.
Pré-requisitos	Sistemas Operacionais
Disc. recomendadas	—
Carga horária total	60h
Natureza	▷ Teóricas: 30h ▷ Práticas: 30h
Caráter	Optativo
Ementa	Apresentação de problemas reais causados pelo aumento do volume, variedade ou velocidade com que dados são disponibilizados e devem ser processados. Apresentação de modelos de programação em larga escala para programação em lotes, como Mapreduce e ferramentas associadas. Apresentar modelos de programação e ferramentas para processamento de fluxos contínuos de dados. Apresentar as principais arquiteturas para processamento de dados em escala.
Respons. pela oferta	DC (1001515)
Bibliografia Básica	
LIN, J.; DYER, C. Data <i>Data-intensive text processing with MapReduce. Synthesis Lectures on Human Language Technologies.</i> v. 3, n.1, p.1-177, 2010.	
WHITE, T. <i>Hadoop: The definitive guide.</i> O'Reilly Media, Inc., 2012.	
Bibliografia Complementar	
MARZ, N.; WARREN, J. <i>A new paradigm for Big Data. Big Data: Principles and best practices of scalable real-time data systems.</i> Shelter Island: Manning Publications, 2014.	

Título	Tecnologia de Comunicação
Objetivo Geral	Capacitar o estudante nas tecnologias de comunicação para redes de computadores, abordando suas operações, funcionalidades e serviços, incluindo aspectos de hardware dos módulos de transmissão e recepção, em meios guiados (cabo, fibra ótica) e não guiados (ar, água, espaço sideral). Habilitar o conhecimento das tecnologias de transmissão digital de dados por rádio, incluindo sinais, sinalização por ondas de rádio e óticas, os conceitos teóricos para transmissão analógica e digital, os padrões e protocolos industriais existentes e emergentes, necessários a execução de projetos de sistemas de comunicação para redes de computadores sem fio.
Pré-requisitos	Sistemas Operacionais
Disc. recomendadas	—
Carga horária total	60h
Natureza	► Teóricas: 30h ► Práticas: 30h
Caráter	Optativo
Ementa	Propagação de sinal em diferentes meios físicos (guiados e não guiados). Estratégias de codificação. Modulação, Multiplexação. Transmissão Analógica e Digital. Telefonia e Comutação. Topologias de Rede de Computadores. Controle de Acesso ao meio físico. Controle do Enlace de Dados. Delimitações, Endereçamento, Tratamento de Erros e Encapsulamento. Tecnologias e Padrões de Comunicação da Indústria e Emergentes. Padrões de Interoperabilidade e Segurança; Radio definido por Software. Projeto e Implementação de Sistema de Comunicação.
Respons. pela oferta	DC (1001505)
Bibliografia Básica	
NICOLAIDIS, I.; BARBEAU, M.; KRANAKIS, E. <i>Ad-Hoc, Mobile and Wireless Networks</i> . Heidelberg Springer-Verlag, 2004, ISBN 3-540-22543-9.	
STALLINGS, W. <i>Data and Computer Communications</i> . 6. ed. Upper Saddle River: Prentice Hall, 2000, ISBN 0-13-084370-9.	
Bibliografia Complementar	
JOHNSON JR, C.R.; SETHARES, W.A.; <i>Telecommunication Breakdown - Concepts of Communication Transmitted via Software Defined Radio</i> . Pearson-Prentice Hall, 2003, ISBN 0-13-143047-5.	
RAPPORT, T.S.; <i>Comunicações sem Fio, Princípios e Práticas</i> . 2. ed. Pearson-Prentice Hall, 2009, ISBN 978-85-7605-198-5.	

4.5.8 Disciplinas do Oitavo Semestre

Título	Estágio em Computação
Objetivo Geral	Tornar o aluno apto a desempenhar atividades profissionais reais e aplicar os conhecimentos adquiridos no curso na prática
Pré-requisitos	Ter sido aprovado em no mínimo 1830 horas
Disc. recomendadas	N/A
Carga horária total	360h
Natureza	▷ Estágio: 360h
Caráter	Obrigatório
Ementa	Desenvolvimento de projetos; Desenvolvimento de sistemas de tipos variados; participação em reuniões; exposição a contextos reais de trabalho
Respons. pela oferta	DC (1001531)
Bibliografia Básica	Material disponibilizado pela empresa, caso seja necessário para a complementação da formação do estudante. Masiero, Paulo Cesar. Ética em computação. São Paulo: EDUSP, 2000. 213 p. ISBN 85-314-0575-0
Bibliografia Complementar	Material disponibilizado pela empresa, caso seja necessário para a complementação da formação do estudante.

Título	Trabalho de Conclusão de Curso
Objetivo Geral	Aplicar os conhecimentos adquiridos no Curso e adquirir novos conhecimentos através de trabalhos práticos desenvolvidos na UFSCar.
Pré-requisitos	Ter sido aprovado em no mínimo 2310 horas
Disc. recomendadas	N/A
Carga horária total	360h
Natureza	► Teóricas: 360h
Caráter	Obrigatório
Ementa	Elaboração de um projeto para o trabalho de conclusão de curso sob a orientação de um docente.
Respons. pela oferta	DC (1001523)
Bibliografia Básica	Bibliografia básica de acordo com o projeto estabelecido junto ao orientador.
Bibliografia Complementar	Bibliografia complementar de acordo com o projeto estabelecido junto ao orientador.

4.6 Dispensas entre Matrizes Curriculares

Esta seção apresenta a relação de dispensa entre as disciplinas da matriz antiga e da matriz curricular deste projeto pedagógico. A relação de dispensa é unidirecional, isto é, só vale de um lado. Por exemplo, suponha que a disciplina A2 é dispensada pela disciplina A1. Isso significa que se um estudante foi aprovado na disciplina A1 ele não precisa cursar A2, mas o oposto não é verdade. A dispensa sempre é do ponto de vista do estudante que migrou de grade, isto é, a intenção é mostrar quais disciplinas da nova matriz são dispensas por disciplinas da matriz antiga. Maiores detalhes sobre dispensas podem ser obtidas no Regimento Geral dos Cursos de Graduação da UFSCar (UNIVERSIDADE FEDERAL DE SÃO CARLOS - UFSCAR, Setembro de 2016).

Os quadros abaixo são divididos da seguinte forma. O Quadro 4.16 concentra as disciplinas obrigatórias do curso e exibe, do lado esquerdo, as disciplinas da matriz que está sendo reformulada e, na coluna do lado direito, as disciplinas da matriz antiga que dispensam as novas.

Os Quadros 4.17, 4.18 e 4.19 mostram as relações de dispensa entre as disciplinas optativas de linha e as antigas do curso.

Quadro 4.16: Dispensas entre Matrizes (disciplinas obrigatórias) - Leitura: Disciplina Nova é Dispensada por Disciplina Antiga

Matriz Nova						Matriz Antiga							
Perfil	Nome			Horas		Deptº	Perfil	Nome			Horas		Deptº
	T	P	Ex					T	P	Ex			
1	Construção de Algoritmos e Programação (NOVA)	60	60	0	DC	1	Construção de Algoritmos e Programação (1001350)	60	60	0	DC		
1	Lógica Digital (1003451)	60	30	0	DC	1	Lógica Digital (1001351)	60	30	0	DC		
1	Introdução ao Pensamento Algorítmico (1001349)	30	0	0	DC	1	Introdução ao Pensamento Algorítmico (1001349)	30	0	0	DC		
1	Matemática Discreta (NOVA)	60	0	0	DC	3	Matemática Discreta (1001500)	60	0	0	DC		
1	Cálculo Diferencial e Integral 1 (1003500)	75	15	0	DM	1	Cálculo Diferencial e Integral 1 (82210)	75	15	0	DM		
2	Lógica Matemática (1001529)	60	0	0	DC	2	Lógica Matemática (1001529)	60	0	0	DC		
2	Geometria Analítica (1003457)	45	15	0	DM	2	Geometria Analítica (81116)	45	15	0	DM		
2	Programação Orientada a Objetos (1001507)	30	30	0	DC	2	Programação Orientada a Objetos (1001507)	30	30	0	DC		
2	Algoritmos e Estruturas de Dados 1 (NOVA)	60	0	0	DC	2	Algoritmos e Estruturas de Dados 1 (1001502)	60	0	0	DC		
2	Arq. e Org. de Computadores 1 (NOVA)	60	30	0	DC	2	Arq. e Org. de Computadores 1 (1001540)	60	30	0	DC		

2	Estatística Básica (1003522)	60	0	0	DES	2	Probabilidade e Estatística (150010)	60	0	0	DES
3	Cálculo Diferencial e Séries (82260)	45	15	0	DM	3	Cálculo Diferencial e Séries (82260)	45	15	0	DM
3	Algoritmos e Estruturas de Dados 2 (NOVA)	60	0	0	DC	3	Algoritmos e Estruturas de Dados 2 (1001490)	60	0	0	DC
3	Metodologia Científica (NOVA)	60	0	0	DC	7	Metodologia Científica (1001343)	60	0	0	DC
3	Computação e Sociedade (NOVA)	0	0	60	DC	—	Não há dispensa	—	—	—	DC
3	Organização e Recuperação da Informação (1001487)	60	0	0	DC	4	Organização e Recuperação da Informação (1001487)	60	0	0	DC
3	Sistemas Operacionais (1001535)	60	30	0	DC	3	Sistemas Operacionais (1001535)	60	30	0	DC
4	Álgebra Linear 1 (1003501)	45	15	0	DM	4	Álgebra Linear 1 (80136)	45	15	0	DM
4	Engenharia de Software 1 (1001530)	60	0	0	DC	4	Engenharia de Software 1 (1001530)	60	0	0	DC
4	Projeto e Análise de Algoritmos (NOVA)	60	0	0	DC	4	Projeto e Análise de Algoritmos (1001525)	60	0	0	DC
4	Banco de Dados (1001493)	30	30	0	DC	4	Banco de Dados (1001493)	30	30	0	DC
4	Inteligência Artificial (1001336)	30	30	0	DC	4	Inteligência Artificial (1001336)	30	30	0	DC
4	Interação Humano-Computador (1001508)	30	30	0	DC	4	Interação Humano-Computador (1001508)	30	30	0	DC
5	Cálculo Numérico (NOVA)	45	15	0	DM	5	Cálculo Numérico (83020)	45	15	0	DM
5	Redes de Computadores (1001504)	60	0	0	DC	5	Redes de Computadores (1001504)	60	0	0	DC
6	Teoria da Computação (1001510)	60	0	0	DC	6	Teoria da Computação (1001510)	60	0	0	DC
6	Programação Paralela e Distribuída (1001483)	30	30	0	DC	6	Programação Paralela e Distribuída (1001483)	30	30	0	DC
7	Construção de Compiladores (1001497)	60	0	0	DC	7	Construção de Compiladores (1001497)	60	0	0	DC

Quadro 4.17: Dispensas: Disciplinas Optativas do Semestre 5 que são dispensadas por Antigas

Matriz Nova				Matriz Antiga				
Nome		Horas		Nome		Horas		
		T	P	Ex				
Program. Orientada a Objetos Avançada (1001521)		30	30	0	Program. Orientada a Objetos Avançada (1001521)	30	30	0
Desenvolvimento de Software para Web 1 (1001542)	0	60	0	Desenvolvimento de Software para Web 1 (1001542)	0	60	0	
Engenharia de Software 2 (1001516)	60	0	0	Engenharia de Software 2 (1001516)	60	0	0	
Proj. e Implementação de Banco de Dados (1001546)	30	30	0	Proj. e Implementação de Banco de Dados (1001546)	30	30	0	
Aprendizado de Máquina 1 (1001524)	60	0	0	Aprendizado de Máquina 1 (1001524)	60	0	0	
Computação Gráfica (1001536)	30	30	0	Computação Gráfica (1001536)	30	30	0	
Processamento Digital de Imagens (1001527)	30	30	0	Processamento Digital de Imagens (1001527)	30	30	0	
Sistemas Distribuídos (1001503)	60	0	0	Sistemas Distribuídos (1001503)	60	0	0	
Processamento de Sinais Digitais (1001486)	60	30	0	Processamento de Sinais Digitais (1001486)	60	30	0	
Projeto e Análise de Algoritmos 2 (NOVA)	60	0	0	Não há dispensa	—	—	—	

Quadro 4.18: Dispensas: Disciplinas Optativas do Semestre 6 que são dispensadas por Antigas

Matriz Nova				Matriz Antiga			
Nome	Horas			Nome	Horas		
	T	P	Ex		T	P	Ex
Paradigmas de Linguagens de Programação (1001511)	30	30	0	Paradigmas de Linguagens de Programação (1001511)	30	30	0
Desenvolvimento de Software para Web 2 (1001543)	0	60	0	Desenvolvimento de Software para Web 2 (1001543)	0	60	0
Arquitetura de Software e Padrões (1001526)	60	0	0	Arquitetura de Software e Padrões (1001526)	60	0	0
Banco de Dados para Ciência de Dados (1001512)	30	30	0	Banco de Dados para Ciência de Dados (1001512)	30	30	0
Aprendizado de Máquina 2 (1001513)	60	0	0	Aprendizado de Máquina 2 (1001513)	60	0	0
Processamento e Visualização de Dados (1001514)	30	30	0	Processamento e Visualização de Dados (1001514)	30	30	0
Processam. Digital de Imagens 3D e Vídeos (1001480)	30	30	0	Processam. Digital de Imagens 3D e Vídeos (1001480)	30	30	0
Arq. e Organização de Computadores 2 (1001541)	30	30	0	Arq. e Organização de Computadores 2 (1001541)	30	30	0
Sistemas Digitais (1001539)	30	30	0	Sistemas Digitais (1001539)	30	30	0
Segurança Cibernética (1001520)	30	30	0	Segurança Cibernética (1001520)	30	30	0
Otimização Matemática (1001346)	60	0	0	Otimização Matemática (1001346)	60	0	0

Quadro 4.19: Dispensas: Disciplinas Optativas do Semestre 7 que são dispensadas por Antigas

Matriz Nova				Matriz Antiga			
Nome	Horas			Nome	Horas		
	T	P	Ex		T	P	Ex
Desenvolvimento Móvel (1001337)	0	60	0	Desenvolvimento Móvel (1001337)	0	60	0
DevOps (1001544)	0	60	0	DevOps (1001544)	0	60	0
Visão Computacional (1001532)	30	30	0	Visão Computacional (1001532)	30	30	0
Projeto de Sistemas Computacionais Embarcados (1001538)	0	60	0	Projeto de Sistemas Computacionais Embarcados (1001538)	0	60	0
Arquiteturas de Alto Desempenho (1001537)	30	30	0	Arquiteturas de Alto Desempenho (1001537)	30	30	0
Processamento de Dados em Escala (1001515)	30	30	0	Processamento de Dados em Escala (1001515)	30	30	0
Processamento de Linguagem Natural (1003315)	60	0	0	Processamento de Linguagem Natural (1003315)	60	0	0
Tecnologia de Comunicação (1001505)	30	30	0	Tecnologia de Comunicação (1001505)	30	30	0

4.7 Plano de migração para novo currículo

De acordo com o artigo 84 do Regimento Geral dos Cursos de Graduação da UFSCar, na implantação de um novo currículo, a opção pelo novo currículo é facultada aos estudantes que ainda não tiverem concluído 50% (cinquenta por cento) da carga horária total do curso. Assim, os estudantes que tenham cumprido até 1.620 horas de um total de 3.240 horas, estão aptos para migração.

Considerando-se as dispensas de disciplinas apresentadas, sugere-se na Seção 4.7.1 uma matriz curricular para integralização do curso para os alunos que estejam enquadrados no perfil de 3º período em 2026/1. Na Seção 4.7.2, sugere-se uma matriz curricular para integralização do curso para os alunos que estejam enquadrados no perfil de 5º período em 2026/1.

Alunos fora do perfil devem avaliar as condições para a migração de plano.

4.7.1 Plano de migração para alunos enquadrados no 3º período em 2026/1

Os alunos enquadrados no 3º período podem ser dispensados das disciplinas da nova matriz curricular apresentadas no Quadro 4.20.

Quadro 4.20: Dispensas para alunos no 3º período

Matriz Nova				Matriz Antiga					
Perfil	Nome	Horas			Perfil	Nome	Horas		
		T	P	Ex			T	P	Ex
1	Construção de Algoritmos e Programação (NOVA)	60	60	0	1	Construção de Algoritmos e Programação (1001350)	60	60	0
1	Lógica Digital (1003451)	60	30	0	1	Lógica Digital (1003451)	60	30	0
1	Introdução ao Pensamento Algorítmico (1001349)	30	0	0	1	Introdução ao Pensamento Algorítmico (1001349)	30	0	0
1	Cálculo Diferencial e Integral 1 (1003500)	75	15	0	1	Cálculo Diferencial e Integral 1 (82210)	75	15	0
2	Lógica Matemática (1001529)	60	0	0	2	Lógica Matemática (1001529)	60	0	0
2	Geometria Analítica (1003457)	45	15	0	2	Geometria Analítica (81116)	45	15	0
2	Programação Orientada a Objetos (1001507)	30	30	0	2	Programação Orientada a Objetos (1001507)	30	30	0
2	Algoritmos e Estruturas de Dados 1 (NOVA)	60	0	0	2	Algoritmos e Estruturas de Dados 1 (1001502)	60	0	0
2	Arq. e Org. de Computadores 1 (NOVA)	60	30	0	2	Arq. e Org. de Computadores 1 (1001540)	60	30	0
2	Estatística Básica (1003522)	60	0	0	2	Probabilidade e Estatística (150010)	60	0	0

Considerando-se as dispensas de disciplinas apresentadas, sugere-se a seguir uma grade curricular para integralização do curso para os alunos que estejam enquadrados no perfil de 3º período em 2026/1.

Quadro 4.21: 3º Semestre - 2026/1 (450 horas).

nº	Disciplina	Requisitos	Deptº. Ofertante	Caráter	Natureza (horas)				
					T	P	Ex	Es	Total
1	Cálculo Diferencial e Séries	Cálculo 1	DM	Obrigatório	45	15	—	—	60
2	Algoritmos e Estruturas de Dados 2	Algoritmos e Estruturas de Dados 1	DC	Obrigatório	60	—	—	—	60
3	Organização e Recuperação da Informação	Algoritmos e Estruturas de Dados 1	DC	Obrigatório	60	—	—	—	60
4	Sistemas Operacionais	Arquitetura e Organização de Computadores 1	DC	Obrigatório	60	30	—	—	90
5	Computação e Sociedade	—	DC	Obrigatório	—	—	60	—	60
6	Metodologia Científica	—	DC	Obrigatório	60	—	—	—	60
7	Matemática Discreta	—	DC	Obrigatório	60	—	—	—	60

127

Quadro 4.22: 4º Semestre - 2026/2 (360 horas).

nº	Disciplina	Requisitos	Deptº. Ofertante	Caráter	Natureza (horas)				
					T	P	Ex	Es	Total
1	Álgebra Linear 1	Geometria Analítica	DM	Obrigatório	45	15	—	—	60
2	Projeto e Análise de Algoritmos	Algoritmos e Estruturas de Dados 2	DC	Obrigatório	60	—	—	—	60
3	Interação Humano-Computador	Construção de Algoritmos e Programação	DC	Obrigatório	30	30	—	—	60
4	Engenharia de Software 1	Programação Orientada a Objetos	DC	Obrigatório	60	—	—	—	60
5	Banco de Dados	Algoritmos e Estrutura de Dados 1	DC	Obrigatório	30	30	—	—	60
6	Inteligência Artificial	Algoritmos e Estruturas de Dados 1	DC	Obrigatório	30	30	—	—	60

Quadro 4.23: 5º Semestre - 2027/1 (300 horas).

nº	Disciplina	Requisitos	Deptº. Ofertante	Caráter	Natureza (horas)				
					T	P	Ex	Es	Total
1	Cálculo Numérico	Cálculo Diferencial e Integral 1, Geometria Analítica e Construção de Algoritmos e Programação	DM	Obrigatório	45	15	—	—	60
2	Redes de Computadores	Sistemas Operacionais	DC	Obrigatório	60	—	—	—	60
3	Disciplina Optativa	—	DC	Optativo	—	—	—	—	60
4	Disciplina Optativa	—	DC	Optativo	—	—	—	—	60
5	Disciplina Optativa	—	DC	Optativo	—	—	—	—	60

Quadro 4.24: 6º Semestre - 2027/2 (300 horas).

128

nº	Disciplina	Requisitos	Deptº. Ofertante	Caráter	Natureza (horas)				
					T	P	Ex	Es	Total
1	Teoria da Computação	Matemática Discreta	DC	Obrigatório	60	—	—	—	60
2	Programação Paralela e Distribuída	Sistemas Operacionais	DC	Obrigatório	30	30	—	—	60
3	Disciplina Optativa	—	DC	Optativo	—	—	—	—	60
4	Disciplina Optativa	—	DC	Optativo	—	—	—	—	60
5	Disciplina Optativa	—	DC	Optativo	—	—	—	—	60

Quadro 4.25: 7º Semestre - 2028/1 (240 horas).

nº	Disciplina	Requisitos	Dept. Ofertante	Caráter	Natureza (horas)				
					T	P	Ex	Es	Total
1	Construção de Compiladores	Teoria da Computação e Construção de Algoritmos e Programação	DC	Obrigatório	60	—	—	—	60
2	Disciplina Optativa	—	DC	Optativo	—	—	—	—	60
3	Disciplina Optativa	—	DC	Optativo	—	—	—	—	60
4	Disciplina Optativa	—	DC	Optativo	—	—	—	—	60

Quadro 4.26: 8º Semestre - 2028/2 (escolher entre Estágio ou TCC: 360 horas).

nº	Disciplina	Requisitos	Dept. Ofertante	Caráter	Natureza (horas)				
					T	P	Ex	Es	Total
129	Estágio em Computação	Ter sido aprovado em no mínimo 1830 horas	DC	Obrigatório	—	—	—	360	360
	Trabalho de Conclusão de Curso	Ter sido aprovado em no mínimo 2310 horas	DC	Obrigatório	360	—	—	—	360

As listagens de disciplinas Optativas são as mesmas apresentadas na seção 4.4.

4.7.2 Plano de migração para alunos enquadrados no 5º período em 2026/1

Os alunos enquadrados no 5º período podem ser dispensados das disciplinas da nova matriz curricular apresentadas no Quadro 4.27.

Quadro 4.27: Dispensas para alunos no 5º período

Matriz Nova				Matriz Antiga					
Perfil	Nome	Horas			Perfil	Nome	Horas		
		T	P	Ex			T	P	Ex
1	Construção de Algoritmos e Programação (NOVA)	60	60	0	1	Construção de Algoritmos e Programação (1001350)	60	60	0
1	Lógica Digital (1003451)	60	30	0	1	Lógica Digital (1003451)	60	30	0
1	Introdução ao Pensamento Algorítmico (1001349)	30	0	0	1	Introdução ao Pensamento Algorítmico (1001349)	30	0	0
1	Matemática Discreta (NOVA)	60	0	0	3	Matemática Discreta (1001500)	60	0	0
1	Cálculo Diferencial e Integral 1 (1003500)	75	15	0	1	Cálculo Diferencial e Integral 1 82210	75	15	0
2	Lógica Matemática (1001529)	60	0	0	2	Lógica Matemática (1001529)	60	0	0
2	Geometria Analítica (1003457)	45	15	0	2	Geometria Analítica (81116)	45	15	0
2	Programação Orientada a Objetos (1001507)	30	30	0	2	Programação Orientada a Objetos (1001507)	30	30	0
2	Algoritmos e Estruturas de Dados 1 (NOVA)	60	0	0	2	Algoritmos e Estruturas de Dados 1 (1001502)	60	0	0
2	Arq. e Org. de Computadores 1 (NOVA)	60	30	0	2	Arq. e Org. de Computadores 1 (1001540)	60	30	0
2	Estatística Básica (1003522)	60	0	0	2	Probabilidade e Estatística (150010)	60	0	0
3	Cálculo Diferencial e Séries (82260)	45	15	0	3	Cálculo Diferencial e Séries (82260)	45	15	0
3	Algoritmos e Estruturas de Dados 2 (NOVA)	60	0	0	3	Algoritmos e Estruturas de Dados 2 (1001490)	60	0	0
3	Organização e Recuperação de Informação (1001487)	60	0	0	3	Organização e Recuperação de Informação (1001487)	60	0	0
3	Sistemas Operacionais (1001535)	60	30	0	3	Sistemas Operacionais (1001535)	60	30	0
4	Álgebra Linear 1 (1003501)	45	15	0	4	Álgebra Linear 1 (80136)	45	15	0
4	Projeto e Análise de Algoritmos (NOVA)	60	0	0	4	Projeto e Análise de Algoritmos (1001525)	60	0	0
4	Interação Humano Computador (1001508)	30	30	0	4	Interação Humano Computador (1001508)	30	30	0
4	Engenharia de Software 1 (1001530)	60	0	0	4	Engenharia de Software 1 (1001530)	60	0	0
4	Banco de Dados (1001493)	30	30	0	4	Banco de Dados (1001493)	30	30	0
4	Inteligência Artificial (1001336)	30	30	0	4	Inteligência Artificial (1001336)	30	30	0

Considerando-se as dispensas de disciplinas apresentadas, sugere-se a seguir uma grade curricular para integralização do curso para os alunos que estejam enquadrados no perfil de 5º período em 2026/1.

Quadro 4.28: 5º Semestre - 2026/1 (420 horas).

nº	Disciplina	Requisitos	Deptº. Ofertante	Caráter	Natureza (horas)				
					T	P	Ex	Es	Total
1	Cálculo Numérico	Cálculo Diferencial e Integral 1, Geometria Analítica e Construção de Algoritmos e Programação	DM	Obrigatório	45	15	—	—	60
2	Redes de Computadores	Sistemas Operacionais	DC	Obrigatório	60	—	—	—	60
3	Metodologia Científica	—	DC	Obrigatório	60	—	—	—	60
4	Computação e Sociedade	—	DC	Obrigatório	—	—	60	—	60
5	Disciplina Optativa	—	DC	Optativo	—	—	—	—	60
6	Disciplina Optativa	—	DC	Optativo	—	—	—	—	60
7	Disciplina Optativa	—	DC	Optativo	—	—	—	—	60

131

Quadro 4.29: 6º Semestre - 2026/2 (300 horas).

nº	Disciplina	Requisitos	Deptº. Ofertante	Caráter	Natureza (horas)				
					T	P	Ex	Es	Total
1	Teoria da Computação	Matemática Discreta	DC	Obrigatório	60	—	—	—	60
2	Programação Paralela e Distribuída	Sistemas Operacionais	DC	Obrigatório	30	30	—	—	60
3	Disciplina Optativa	—	DC	Optativo	—	—	—	—	60
4	Disciplina Optativa	—	DC	Optativo	—	—	—	—	60
5	Disciplina Optativa	—	DC	Optativo	—	—	—	—	60

Quadro 4.30: 7º Semestre - 2027/1 (240 horas).

nº	Disciplina	Requisitos	Dept. Ofertante	Caráter	Natureza (horas)				
					T	P	Ex	Es	Total
1	Construção de Compiladores	Teoria da Computação e Construção de Algoritmos e Programação	DC	Obrigatório	60	—	—	—	60
2	Disciplina Optativa	—	DC	Optativo	—	—	—	—	60
3	Disciplina Optativa	—	DC	Optativo	—	—	—	—	60
4	Disciplina Optativa	—	DC	Optativo	—	—	—	—	60

Quadro 4.31: 8º Semestre - 2027/2 (escolher entre Estágio ou TCC: 360 horas).

nº	Disciplina	Requisitos	Dept. Ofertante	Caráter	Natureza (horas)				
					T	P	Ex	Es	Total
1	Estágio em Computação	Ter sido aprovado em no mínimo 1830 horas	DC	Obrigatório	—	—	—	360	360
2	Trabalho de Conclusão de Curso	Ter sido aprovado em no mínimo 2310 horas	DC	Obrigatório	360	—	—	—	360

As listagens de disciplinas Optativas são as mesmas apresentadas na seção 4.4.

5. Integração Ensino, Pesquisa e Extensão

No âmbito deste projeto pedagógico, o termo “ensino” envolve atividades relacionadas ao processo de ensino/aprendizagem, fazendo com que o aluno adquira novos conhecimentos. O termo “pesquisa” se refere a atividades que produzem avanço no conhecimento científico, incluindo inovação. Enquanto a “extensão”, segundo a Resolução Conjunta CoG 2/2023: “Extensão Universitária constitui-se em processo interdisciplinar, político educacional, cultural, científico, tecnológico, que promove a interação transformadora entre as instituições de ensino superior e os outros setores da sociedade, por meio da produção e da aplicação do conhecimento, em articulação permanente com o ensino e a pesquisa.”

Em um curso de excelência, a integração entre ensino, pesquisa e extensão é fundamental para que a universidade cumpra seu papel na sociedade. Com o ensino, a universidade desenvolve competências e habilidades no estudante, que quando formado, as utilizará na sociedade, melhorando-a. O ensino também fornece ao estudante a fundamentação teórica necessária para a realização de pesquisa formando, assim, pesquisadores competentes.

Com a pesquisa, promove-se um constante aperfeiçoamento dos professores, fazendo que seu conhecimento sobre um assunto seja aprofundado, culminando em um ensino de melhor qualidade. A pesquisa também gera conhecimento que pode ser transferido para resolução de problemas reais ou promover melhorias na sociedade.

Com a extensão, obtém-se um contato mais próximo entre a universidade e a sociedade. O ensino é beneficiado, pois com a extensão é possível trazer para a universidade demandas reais da sociedade, que podem ser incorporadas aos componentes curriculares. Também beneficia a pesquisa, pois as demandas e oportunidades estimulam a produção de conhecimento que pode ser mais diretamente aplicado na sociedade, produzindo inovação.

Cada componente curricular do Bacharelado em Ciência da Computação proporciona um nível diferente de integração entre ensino, pesquisa e extensão, como pode ser visto no Quadro a seguir. Com a inserção da extensão nas atividades curriculares, a partir deste Projeto Pedagógico, ela passa a figurar de modo mais presente nos componentes do curso. As atividades complementares são uma exceção, visto que este curso oferece uma grande diversidade de atividades complementares que podem ter caráter de ensino, pesquisa, extensão ou inclusive vários deles. Detalhes sobre as atividades complementares podem ser obtidos no Apêndice IV.

Componente curricular	Ensino	Pesquisa	Extensão
Disciplinas	X		X
Áreas de conhecimento	X		
Projeto Integrador Extensionista	X	X	X
Atividades Curriculares de Extensão	X		X
TCC	X	X	
Estágio	X		X
Atividades Complementares	X	X	X

Para a consecução do perfil do egresso idealizado, a abordagem multi/interdisciplinar figura como fundamental para a geração integrada de conhecimento, cuja prática e contato com os problemas reais serão vivenciados pelo egresso no exercício profissional ou pelo estudante durante seu Estágio. Neste sentido, o Estágio Supervisionado, regulamentado pela Lei nº 11.788, de 25 de setembro de 2008, prevê uma carga horária de no mínimo de 360 horas, realizado no 8º período do curso. Outro importante momento de vivência prática são as aulas em laboratório, posto que nelas são propiciadas as simulações de situações reais em ambiente acadêmico, instigando o estudante à observação e compreensão dos diversos fenômenos reproduzidos.

Outro aspecto relevante e vinculado ao desenvolvimento das competências e habilidades delineadas neste Projeto Pedagógico se refere ao estímulo para a realização de trabalhos em equipe, na medida em que estes promovem a troca de informações, implicando na organização do trabalho a partir da divisão de tarefas e compartilhamento de responsabilidades. O Cientista da Computação formado na UFSCar possui uma sólida formação em algumas ciências básicas, como a Matemática e a Matemática Computacional. Essa base teórica está sempre integrada ao desenvolvimento de soluções práticas, permitindo ao estudante compreender e modelar problemas

complexos. O curso estimula a criatividade e prepara o egresso para projetar, implementar e aplicar metodologias computacionais inovadoras na resolução de desafios reais, unindo rigor matemático e aplicação tecnológica. Na medida do possível, as disciplinas com alto conteúdo tecnológico são e serão desenvolvidas com base na discussão e resolução de problemas reais das empresas.

O curso de Ciência da Computação é direcionado ao estudante com uma forte aptidão para ciências exatas e que deseja aplicar esses conhecimentos básicos na investigação e na resolução de problemas tecnológicos. Assim, não se considera, aqui, a histórica divisão entre as especializações e ataca-se o problema proposto por meio de uma estratégia multidisciplinar, tornando a formação mais que generalista: uma formação de multiespecialista.

A possibilidade que o estudante de Ciência da Computação tem, ao frequentar e participar de atividades relacionadas aos diferentes grupos de pesquisa do Departamento de Computação, ou de outros departamentos da UFSCar, consiste em uma oportunidade ímpar na sua formação. Neste sentido, algumas das experiências relacionadas às disciplinas práticas são realizadas no âmbito desses grupos.

5.1 Articulação Ensino, Pesquisa e Extensão

A UFSCar, ao longo de sua história, preocupa-se em promover ativamente a integração entre as atividades de ensino, pesquisa e extensão, reconhecendo que essas atividades, quando adequadamente articuladas e executadas de forma balanceada, potencializam-se umas às outras. Esta diretriz acadêmica também está fundamentada neste projeto, já que os estudantes poderão se envolver com atividades de ensino, pesquisa e extensão, vinculadas diretamente ao curso ou ofertadas pelos Departamentos com ele comprometidos.

5.2 Atividades de Pesquisa

Uma primeira estratégia para desenvolver no estudante as habilidades de pesquisa no curso de Ciência da Computação é a oportunidade de realizar a Iniciação Científica (IC). Para tanto, a IC deve ser formalizada pelos canais internos da universidade, conforme sua afinidade temática, em conjunto com o docente orientador. Uma iniciação científica é uma atividade desenvolvida pelo

estudante sob a supervisão de um professor orientador. O objetivo é que o aluno desenvolva algum projeto que, normalmente, demanda 1 ano e que equivale a 165 horas de atividade complementar. A IC permite que o estudante tenha contato com temas inovadores da computação e que tenha a experiência do que significa fazer pesquisa em computação. Em muitos casos, o estudante acaba tendo contato direto com os alunos de mestrado e doutorado do departamento e desenvolve seu projeto em laboratórios de pesquisa. Uma iniciação científica é bastante relevante para estudantes que pretendem seguir carreira acadêmica e de pesquisa, geralmente fazendo mestrado e doutorado. Mas, atualmente, também há muitas empresas que valorizam egressos com habilidades de pesquisa.

Outra forma de articulação com pesquisa é o Projeto Integrador Extensionista. Esse projeto visa a conciliar as habilidades já adquiridas pelos estudantes para a resolução de problemas de interesse da sociedade e que, ao mesmo tempo, apresentem características de inovação, exigindo dos mesmos uma dedicação à investigação do estado da arte para a resolução de tais problemas.

Outra forma ainda de contato com pesquisa são os grupos PET (Programa de Educação Tutorial). O Bacharelado em Ciência da Computação possui um grupo PET que tem o objetivo de desenvolver atividades de ensino, pesquisa e extensão. Diferentemente de uma iniciação científica, estudantes petianos tem contato com os três pilares da universidade (ensino, pesquisa e extensão). As pesquisas conduzidas dentro do grupo PET são menos profundadas do que pesquisas que são feitas na forma de iniciações científicas, já que essas possuem a pesquisa como norte prioritário.

5.3 Atividades de Extensão

As atividades de extensão são importantes não apenas como meio de difusão do conhecimento gerado na universidade, mas também como mecanismo de aproximação da realidade. De maneira mais explícita, os estudantes terão a oportunidade de participar de atividades de extensão organizadas pelos diversos canais internos da universidade. A Universidade Federal de São Carlos valoriza estas atividades e tem, na Pró-Reitoria de Extensão, um órgão da sua administração central totalmente dedicado à organização e ao desenvolvimento de atividades de extensão, inclusive financiando parcialmente estas iniciativas.

A diversidade das atividades de pesquisa e extensão beneficia os estudantes de graduação que se envolvem diretamente com elas em projetos de extensão, alargando sua formação com atividades complementares. Dentre as várias iniciativas presentes na Universidade Federal de São Carlos, destaca-se o movimento das empresas juniores (Empresa Jr.) e os grupos de competição em olimpíadas de informática e maratona de programação.

A Ciência da Computação possui sua empresa júnior (compartilhada com o Curso Engenharia de Computação), sediada no Departamento de Computação da UFSCar e que desenvolve diversos projetos junto a empresas da região e a comunidade acadêmica.

Visando incentivar ainda mais a participação dos estudantes na solução de problemas reais e que afetam a sociedade, o curso de Ciência da Computação instituiu o Projeto Integrador Extensionista. Esse tipo de atividade de extensão, detalhado no Apêndice V, caracteriza-se por permitir aos estudantes aplicarem conceitos adquiridos em várias disciplinas do curso na solução de problemas reais demandados pela comunidade externa.

As Atividades Curriculares de Extensão previstas para este curso são detalhadas no Apêndice III. Vale enfatizar que 10% da carga horária do curso contempla extensão.

6. Avaliação da Aprendizagem

Este capítulo descreve os princípios gerais e as formas de avaliação da aprendizagem adotados no curso de Bacharelado em Ciência da Computação, seguindo os preceitos do Regimento Geral de Cursos de Graduação da UFSCar.

6.1 Princípios gerais de avaliação da aprendizagem

A avaliação da aprendizagem, concebida como um processo contínuo de acompanhamento do desempenho dos estudantes, é feita por meio de procedimentos, instrumentos e critérios adequados aos objetivos, conteúdos e metodologias relativas a cada componente curricular. É um elemento essencial de reordenação da prática pedagógica, pois permite um diagnóstico da situação e indica formas de intervenção no processo, com vistas ao desenvolvimento das competências desejadas para o egresso, e promovendo aprendizagem e reflexão sobre a própria prática, tanto para os estudantes como para os professores.

Compreender a avaliação como diagnóstico significa ter o cuidado constante de observar, nas produções e manifestações dos estudantes, os sinais ou indicadores de sua situação de aprendizagem. Na base desta avaliação está o caráter contínuo de diagnóstico e acompanhamento, sempre tendo em vista o progresso dos estudantes e sua aproximação aos alvos pretendidos a partir de sua situação real.

A avaliação presente no curso de Bacharelado em Ciência da Computação da UFSCar está fundamentada na concepção de que o que se pretende não é simplesmente medir aprendizagem segundo escalas ou valores, mas interpretar a caminhada dos estudantes com base nos registros e apreciações sobre seu trabalho. Além disso, segue normas internas sem, no entanto, limitar a liberdade de cada professor. As avaliações são realizadas em vários momentos e não se restringem

somente a uma avaliação de conteúdos. Há avaliações em grupo e individuais, trabalhos, listas de exercícios, participação, interesse, pontualidade e assiduidade.

Entendida desta maneira, a avaliação só tem sentido quando articulada ao projeto pedagógico institucional, que lhe confere significado, e enquanto elemento constituinte do processo educativo, como instrumento que objetiva determinar novos rumos ou corrigir o rumo atual. No que se refere aos aspectos administrativos presentes na sistemática de avaliação de rendimento dos estudantes, o Bacharelado em Ciência da Computação da UFSCar segue os preceitos dos artigos de 18 a 28 do Regimento Geral dos Cursos de Graduação da UFSCar (UNIVERSIDADE FEDERAL DE SÃO CARLOS - UFSCAR, Setembro de 2016), homologado pela Resolução ConsUni nº 867, de 27 de outubro de 2016.

Em síntese, a avaliação presente no curso de Bacharelado em Ciência da Computação da UFSCar tem as seguintes funções:

- acompanhar o desenvolvimento das disciplinas do curso e diagnosticar aspectos que devem ser mantidos ou reformulados em cada uma delas;
- desenvolver, entre os docentes e discentes, uma postura favorável à avaliação, enquanto instrumento das práticas educativas;
- focalizar a produção do conhecimento crítico e transformador;
- avaliar não apenas o conhecimento adquirido, mas também as competências profissionais, por meio do desenvolvimento de trabalhos, projetos, estágios etc;
- avaliar a formação para aptidões específicas e aptidões gerais, envolvendo habilidades práticas e competências para o trabalho em equipe.

Vale ressaltar que essas funções são desempenhadas pela Coordenação de Curso com o apoio dos coordenadores de núcleo de conhecimento, os quais exercem um acompanhamento direto e constante com os docentes responsáveis pelas disciplinas do núcleo que gerenciam.

6.2 Formas de Avaliação da Aprendizagem

A UFSCar, por meio do Regimento Geral dos Cursos de Graduação de setembro de 2016, estabeleceu normas para a sistemática de avaliação do desempenho dos estudantes, estabelecendo que a avaliação é parte integrante e indissociável do ato educativo e deve vincular-se, necessariamente,

ao processo de “ação-reflexão-ação”, que compreende o ensinar e o aprender nas disciplinas e atividades curriculares dos cursos, na perspectiva de *formar profissionais cidadãos capazes de uma ação interativa e responsável na sociedade atual*, caracterizada por sua constante transformação.

A avaliação contínua propicia o acompanhamento da evolução do estudante, bem como através desta se torna possível diagnosticar o conhecimento prévio dos estudantes, refletir sobre os resultados obtidos e construir estratégias de ensino individuais ou coletivas de superação das dificuldades apresentadas.

De acordo com o Artigo 19 do Regimento Geral dos Cursos de Graduação, os Planos de Ensino das disciplinas devem descrever, de forma minuciosa, os procedimentos, instrumentos e critérios de avaliação, diferenciados e adequados aos objetivos, conteúdos e metodologias relativas a cada disciplina. É necessário proporcionar aos estudantes instrumentos de avaliação diferenciados e adequados aos objetivos, multiplicando as suas oportunidades de aprendizagem e diversificando os métodos utilizados. Assim, permite-se que os estudantes apliquem os conhecimentos que adquirem, exercitem e controlem eles próprios a aprendizagem e o desenvolvimento das competências, recebendo comentários e opiniões frequentes dos professores sobre as dificuldades e progressos alcançados.

O Regimento Geral dos Cursos de Graduação prevê, no item II do Artigo 19, a aplicação de procedimentos/instrumentos de avaliações em, pelo menos, três datas distribuídas no período letivo para cada atividade curricular, cabendo ao professor divulgar dois terços dos resultados dos instrumentos aplicados até trinta dias antes do final do período letivo.

A escolha dos métodos e instrumentos de avaliação depende de vários fatores: das finalidades, do objeto de avaliação, da área disciplinar, do tipo de atividade, do contexto, e dos próprios avaliadores. Propõe-se que, além da prova individual com questões dissertativas, outras formas de avaliação sejam consideradas, tais como:

- Trabalhos individuais ou coletivos;
- Atividades de culminância (projetos, monografias, seminários, exposições etc).

Tais instrumentos são utilizados como critério de aprovação do estudante, de acordo com o Artigo 20 do Regimento Geral dos Cursos de Graduação, como segue:

Art. 20. O estudante regularmente inscrito em atividades curriculares é considerado aprovado quando obtiver, simultaneamente:

I - Frequência igual ou superior a setenta e cinco por cento das aulas e/ou das atividades acadêmicas curriculares efetivamente realizadas;

II - Desempenho mínimo equivalente à nota final igual ou superior a 6 (seis) ou conceito equivalente.

Outro aspecto relevante do Regimento Geral dos Cursos de Graduação trata sobre o Processo de Avaliação Complementar (PAC), estabelecido pelo Artigo 22 do regimento. Caso o estudante não obtenha nota final suficiente para sua aprovação, o PAC poderá ser utilizado como recurso para recuperação de conteúdos. Para isso, é necessário que o estudante obtenha frequência igual ou superior a setenta e cinco por cento e nota final igual ou superior a cinco.

O Regimento Geral dos Cursos de Graduação ainda define os prazos para realização do PAC, conforme segue:

Art. 24. O Processo de Avaliação Complementar (PAC) deve ser realizado em período subsequente ao término do período regular de oferecimento da atividade curricular.

Parágrafo Único. A realização do processo de que trata o caput pode prolongar-se até o 35º (trigésimo quinto) dia letivo do período subsequente para atividades curriculares de duração semestral e até 70º (septuagésimo) dia letivo do período subsequente para atividades curriculares de duração anual, não devendo incluir atividades em horários coincidentes com outras atividades curriculares realizadas pelo estudante.

7. Avaliação e Gerenciamento do Curso

Este capítulo dispõe sobre o gerenciamento do curso, que tem como principal objetivo a coordenação didático-pedagógica, visando a elaboração e a condução do projeto pedagógico do curso em concordância com a política de ensino, pesquisa e extensão da Universidade. A gestão do curso é conduzida pelo Coordenador do Curso com o apoio do Conselho de Coordenação de Curso (que possui a função deliberativa), Núcleo Docente Estruturante e Coordenadores de Núcleo de Conhecimento.

7.1 Composição e Funcionamento do Conselho de Coordenação do Curso

O Curso de Ciência da Computação, assim como todos os demais cursos da Universidade Federal de São Carlos, tem sua administração acadêmica regulamentada pelo Regimento Geral dos Cursos de Graduação, que estabelece em seu artigo 87:

Art. 87. A gestão do Curso de Graduação é realizada pelos seguintes órgãos:

- I - Conselho de Coordenação;
- II - Coordenação do Curso.

O Conselho de Coordenação deve ser composto por docentes, servidores técnico-administrativos e estudantes, além dos seus respectivos suplentes. Tal composição é determinada pelo Artigo 89 do Regimento Geral dos Cursos de Graduação, conforme segue:

Art. 89. Cabe ao Conselho de Coordenação do Curso, na definição de seu Regimento Interno, estabelecer os critérios para participação e procedimentos para eleição de seus membros, respeitando a legislação vigente, garantindo, no mínimo:

- I - O Coordenador do Curso como presidente;
 - II - O Vice-Cordenador do Curso como vice-presidente;
 - III - Representação docente dos diversos núcleos de conhecimento ou campos de atuação que compõem o currículo do curso para mandato de dois anos, permitida uma recondução;
 - IV - Representação discente para mandato de um ano, permitida uma recondução.
- §1º. No impedimento do Coordenador e do Vice-Cordenador, a presidência do Conselho de Coordenação de Curso de Graduação é exercida por um docente membro do Conselho de Coordenação, previamente designado pelo Coordenador.
- §2º. Os representantes dos docentes e dos discentes são indicados por seus pares.

Por sua vez, a composição da Coordenação do Curso é definida pelo Artigo 90 do Regimento Geral das Coordenações de Cursos de Graduação, como segue:

- Art. 90. A Coordenação de Curso de Graduação é composta por:
- I - Coordenador de Curso;
 - II - Vice-Cordenador de Curso;
 - III - Secretário de Curso.
- §1º. Cabe ao Coordenador superintender e coordenar as atividades do Curso de Graduação, de acordo com as diretrizes do Conselho de Coordenação.
- §2º. Cabe ao Vice-Cordenador substituir o Coordenador do Curso de Graduação em suas faltas e impedimentos.
- §3º. No impedimento do Coordenador e/ou do Vice-Cordenador, as funções da Coordenação de Curso de Graduação são atribuídas a um docente membro do Conselho de Coordenação, previamente designado pelo Coordenador.

O Conselho de Coordenação de Curso é o responsável por deliberar sobre as ações relativas ao Curso, seguindo as normas estabelecidas pelo Regulamento Geral das Coordenações de Cursos de Graduação da UFSCar. Especificamente, o Artigos 93 e 94 do Regimento definem as competências do Conselho e da Coordenação do curso, respectivamente.

7.2 Núcleo Docente Estruturante

O Núcleo Docente Estruturante (NDE), regido pelos Artigos 98 a 110 do Regimento Geral dos Cursos de Graduação da UFSCar, é um órgão consultivo e propositivo responsável pelo processo de concepção, avaliação e atualização do Projeto Pedagógico do Curso. Para isso, o NDE é responsável pela análise dos resultados de avaliações internas e externas, a fim de propor melhorias ao Conselho de Coordenação do Curso para que o Projeto Pedagógico do Curso possa ser aperfeiçoado.

Além disso, cabe ao NDE zelar pela qualidade da formação do profissional e Diretrizes Curriculares Nacionais para o curso ou legislação correspondente. Ainda, é obrigação do NDE a proposta de atividades de pesquisa e extensão para atender às necessidades da graduação em relação a novas demandas sociais.

O NDE, em conformidade com o Artigo 100 do Regimento Geral dos Cursos de Graduação, é composto por, pelo menos, seis docentes, sendo:

I - Coordenador do Curso;

II - Um mínimo de cinco docentes pertencentes ao corpo docente do curso há pelo menos dois anos.

7.3 O Coordenador de Núcleo de Conhecimento

Conforme abordado na Seção 4.2, o curso possui sete núcleos de conhecimento:

1. Fundamentos de Matemática e Estatística;
2. Fundamentos de Ciência da Computação;
3. Algoritmos e Programação;
4. Metodologia e Técnicas da Computação;
5. Sistemas de Computação
6. Formação Multidisciplinar e Humanística;
7. Orientações;

Cada núcleo de conhecimento possui um **Coordenador de Núcleo de Conhecimento**, que é um professor do curso responsável por promover a integração das disciplinas e atividades

didático-pedagógicas na seu núcleo de responsabilidade. O Coordenador de Núcleo é membro representante de núcleo junto ao NDE. A principal atribuição do Coordenador de Núcleo é zelar pela constante evolução da qualidade do núcleo de conhecimento que ele representa. Para isso, atribuições mais específicas são:

- Promover incrementos e atividades ligadas a seu núcleo de representação para a melhoria do curso como um todo;
- Certificar-se que conteúdos, metodologias de ensino, avaliações e carga horária efetiva de cada disciplina estão sendo cumpridos de acordo com o que estabelece o projeto pedagógico;
- Trabalhar em conjunto com os demais professores dos respectivos núcleos de conhecimento, recomendando-se que ao menos duas reuniões semestrais ocorram: uma de planejamento do semestre a ser iniciado, e outra para avaliação ao final do semestre letivo;
- Reportar semestralmente ao NDE e ao Conselho de Curso sobre as ações realizadas no semestre;
- Zelar por não replicar conteúdo em disciplinas distintas;
- Integrar projetos de diferentes disciplinas da mesma área. Por exemplo, disciplinas em semestres consecutivos podem desenvolver partes de um mesmo projeto; e disciplinas no mesmo semestre podem compartilhar o mesmo projeto e aproveitar a avaliação para as mesmas;
- Integrar docentes e conteúdos de diferentes disciplinas;
- Criar um ambiente de harmonia, colaboração e cooperação entre os docentes do núcleo;
- Promover Atividades Inter-núcleos. As atividades inter-núcleos exigem a comunicação efetiva entre os coordenadores de núcleo para averiguar a viabilidade de desenvolver projetos que envolvam disciplinas de núcleos diferentes. Esse tipo de projeto é o que mais caracteriza a multidisciplinaridade pois poderia, por exemplo, integrar disciplinas do Núcleo de Matemática e Estatística com disciplinas do Núcleo de Metodologia e Técnicas da Computação.
- Auxiliar a integração de diferentes núcleos;
- Indicar formas de incentivo ao desenvolvimento de linhas de pesquisa e extensão.

7.4 Administração e Condução do Curso

O curso de Bacharelado em Ciência da Computação é formado por professores, servidores técnico-administrativos e estudantes e conta com a infra-estrutura disponibilizada pela Pró-reitoria de Graduação da UFSCar (ProGrad) e pelas instalações do Centro de Ciência e Tecnologia da UFSCar (CCET).

Para que o curso realize sua missão de formar estudantes com excelência, é preciso o empenho coletivo de estudantes, docentes e servidores técnico-administrativos (TAs). É imprescindível que todo docente do curso conheça o Projeto Pedagógico e zele pelo seu cumprimento. Com essa atitude o docente terá conhecimento dos princípios pedagógicos que regem o curso. Fica a cargo da chefia do Departamento no qual o docente está lotado, o estímulo dessa prática dentre seus pares.

O NDE e os Coordenadores de Núcleo de Conhecimento devem trabalhar em conjunto, realizando ao menos uma reunião por semestre. A pauta de convocação da reunião deve ser pública e feita com, no mínimo, 48 horas de antecedência. Fica a critério da Coordenação de Curso estabelecer data e horário para que as reuniões ocorram.

7.5 Processo para Autoavaliação do Curso

A avaliação dos cursos de graduação da UFSCar é uma preocupação presente na Instituição e considerada de fundamental importância para o aperfeiçoamento dos projetos pedagógicos dos cursos e a melhoria dos processos de ensino e aprendizagem. Desde a publicação da Lei 10.861 de 14 de abril de 2004, que instituiu o Sistema de Avaliação da Educação Superior (SINAES), a Comissão Própria de Avaliação/UFSCar tem coordenado os processos internos de autoavaliação institucional nos moldes propostos pela atual legislação e contribuído com os processos de avaliação dos cursos. Entre 2023 e 2024, o curso de Ciência da Computação da UFSCar passou por uma avaliação externa realizada pelo INEP/MEC na qual obteve conceito máximo (Conceito 5).

O sistema de avaliação dos cursos de graduação da UFSCar, implantado em 2011, foi concebido pela Pró-Reitoria de Graduação (ProGrad) em colaboração com a Comissão Própria de Avaliação (CPA) com base em experiências institucionais anteriores, quais sejam: o Programa de

Avaliação Institucional das Universidades Brasileiras (PAIUB) e o Programa de Consolidação das Licenciaturas (PRODOCÊNCIA). O PAIUB, iniciado em 1994, realizou uma ampla avaliação de todos os cursos de graduação da UFSCar existentes até aquele momento e o projeto PRODOCÊNCIA/UFSCar, desenvolvido entre os anos de 2007 e 2008, realizou uma avaliação dos cursos de licenciatura dos campi de São Carlos e de Sorocaba.

A avaliação dos cursos de graduação é feita atualmente por meio de formulários de avaliação, os quais são respondidos pelos docentes da área majoritária de cada curso, pelos discentes e, eventualmente, pelos técnico-administrativos e egressos. Esses formulários abordam questões sobre as dimensões do Perfil Profissional a ser formado pela UFSCar; da formação recebida nos cursos; do estágio; da participação em pesquisa, extensão e outras atividades; das condições didático-pedagógicas dos professores; do trabalho das coordenações de curso; do grau de satisfação com o curso realizado; das condições e serviços proporcionados pela UFSCar; e das condições de trabalho para docentes e técnico-administrativos.

A ProGrad, juntamente com a CPA, são responsáveis pela concepção dos instrumentos de avaliação, bem como da divulgação do processo e do encaminhamento dos resultados às respectivas coordenações de curso. Cada Conselho de Coordenação de Curso, bem como seu Núcleo Docente Estruturante (NDE), após o recebimento dos resultados da avaliação, analisam esses resultados para o planejamento de ações necessárias, visando a melhoria do curso.

Deve ser ressaltado que a Coordenação do Curso de Ciência da Computação sempre atuou fortemente não apenas na promoção do curso junto à comunidade externa, mas também no acompanhamento dos egressos. O contato contínuo e intenso com os egressos fornece valiosas informações sobre a colocação dos mesmos no mercado de trabalho e provê informações importantes sobre a formação profissional recebida durante o curso e sua efetividade perante o mercado profissional.

7.6 Apoio ao discente

Desde o ingresso, os estudantes de graduação da UFSCar dispõem de programas de integração que visam facilitar a adaptação à vida acadêmica e fortalecer o senso de pertencimento à comunidade universitária. A semana de recepção aos ingressantes, carinhosamente conhecida

como “Calourada”, promove atividades de integração entre ingressantes e veteranos, bem como fornece informações essenciais para os primeiros passos na vida universitária.

Durante todo o curso, a acessibilidade metodológica e instrumental é assegurada por recursos que atendem às necessidades de estudantes com diferentes perfis, promovendo uma formação inclusiva e equitativa. O curso conta com o apoio da CAAPE (Coordenadoria de Acompanhamento Acadêmico e Pedagógico) que oferece suporte a grupos específicos de estudantes, prioritariamente ingressantes por reserva de vagas, processos seletivos diferenciados e por convênios. Além desse apoio geral da universidade, o curso oferece monitoria e programas de nivelamento, que auxiliam na superação de dificuldades em disciplinas fundamentais, por meio de programas como o PET-BCC e a monitoria e tutoria.

As ações de apoio incluem também iniciativas voltadas à empregabilidade e ao desenvolvimento profissional dos estudantes, por meio da Coordenadora de Estágios e Mobilidade (CEM) da UFSCar e do Coordenador de Estágio. Os estudantes também são incentivados a participar do diretório acadêmico da computação (DAComp), projetos de extensão e atividades de representação estudantil, promovendo o engajamento e a liderança. Mais informações sobre Estágio e Atividades de Extensão podem ser obtidos nos Apêndices I e III.

O curso ainda fomenta a participação em programas de intercâmbio nacionais e internacionais, ampliando a visão de mundo dos estudantes e expondo-os a contextos acadêmicos e culturais diversos. Neste sentido, o curso conta com o apoio da Secretaria Geral de Relações Internacionais (SRInter) da UFSCar.

8. Plano de Implantação

Este capítulo descreve brevemente a infraestrutura física e de pessoal disponível para o bom desenvolvimento do curso de Bacharelado em Ciência da Computação da UFSCar.

8.1 Espaço físico

O curso de Bacharelado em Ciência da Computação (BCC) da UFSCar, campus São Carlos, utiliza a infraestrutura de salas de aula do campus para as aulas teóricas e os laboratórios no prédio do Departamento de Computação (DC) para as aulas práticas. Com 645 hectares de extensão e 196 mil m² de área construída, o Campus São Carlos abriga três centros: o Centro de Ciências Exatas e de Tecnologia (CCET), o Centro de Educação e Ciências Humanas (CECH) e o Centro de Ciências Biológicas e de Saúde (CCBS). Nesses centros estão alocados mais de 30 departamentos acadêmicos que oferecem cerca de 40 cursos de graduação. Os cursos de graduação vinculados ao DC, entre eles o BCC, fazem parte do CCET.

As salas de aula, distribuídas pelos quatro campi da UFSCar, têm dimensões variadas de modo que satisfaçam a proporção média de 1,2 m² por estudante, com ambientes que comportam desde 20 estudantes (com área média de 24 m²), até ambientes maiores, com 180 m² (capacidade de 150 estudantes). O campus São Carlos dispõe de 139 salas de aula que contam com ambientação térmica por meio de ventiladores, iluminação artificial por lâmpadas fluorescentes e iluminação natural por meio de amplas janelas. Os espaços são equipados com carteiras universitárias ou do modelo carteira/cadeira, com mesa e cadeira para o docente e quadro branco ou de giz. Todas as salas possuem equipamentos de multimídia instalados ou disponíveis para empréstimo junto ao setor de zeladoria do prédio.

Todas as salas de aulas são adaptadas à acessibilidade física, dispondo de portas, rampas de acesso e elevadores – conforme o campus – para facilitar o acesso aos andares superiores dos

prédios. As portas são adequadas para o acesso de cadeirante, e o ambiente é projetado para permitir que assistam às aulas com conforto.

As salas de aula recebem manutenção periódica de limpeza programada, conforme sistemática estabelecida entre a Prefeitura Universitária e a empresa terceirizada contratada, responsável pelo serviço, da mesma forma que são analisadas todas as condições de uso da sala, como mobiliário, equipamentos e condições físicas, para eventuais reparos ou substituições, quando necessário.

As salas de aulas teóricas, localizadas em 9 prédios identificados pela sigla AT, são usadas para oferta das atividades curriculares teóricas do BCC, ou seja, que não possuem horas de atividades práticas e, portanto, não demandam o uso de equipamentos de hardware e software.

De uso exclusivo dos estudantes de graduação dos cursos vinculados ao DC, estão 5 laboratórios de ensino (LE-1, LE-2, LE-3, LE-4 e LE-6) e 1 laboratório de informática para uso geral pelos estudantes de graduação, com cerca de 47 m² cada. Dois desses laboratórios (LE-1 e LE-4) são equipados para o ensino relacionado às áreas de arquitetura de computadores, microprocessadores, microcontroladores e lógica digital, contando com equipamentos como osciloscópios, geradores de função e plataformas de prototipagem, além de computadores. Todos os laboratórios possuem cerca de 20 computadores, projetor multimídia, quadro branco e ar-condicionado. A configuração de hardware e software dos laboratórios é atualizada constantemente conforme as demandas das disciplinas práticas.

O DC também conta com um auditório para 80 pessoas (cerca de 122 m²) e um Espaço Maker com capacidade para turmas de 40 estudantes (cerca de 95 m²), ambos equipados com ar-condicionado e projetor multimídia. Em relação ao PPC 2019, o LE-5 foi desativado para dar origem ao Espaço Maker. O Espaço Maker foi inaugurado em 2022 e contém equipamentos como mesas móveis, parede de lousa, bancadas, notebooks, impressoras 3D, máquina de corte a laser, máquina de costura, furadeira, plotter, entre outros. O Espaço Maker é um ambiente dinâmico e criativo que tem se mostrado essencial para a realização de diversos projetos acadêmicos e de pesquisa, permitindo que estudantes e docentes desenvolvam protótipos e soluções inovadoras com recursos modernos, como impressoras 3D, estações de solda, máquinas de corte, entre outros.

As configurações de hardware e software dos laboratórios é atualizada constantemente, e estão disponíveis em <https://www.dc.ufscar.br/sobre-nos/infraestrutura>.

Laboratório	Atividade principal	Capacidade
Laboratório de Ensino 1	Hardware	30 estudantes
Laboratório de Ensino 2	Programação e desenvolvimento	40 estudantes
Laboratório de Ensino 3	Programação e desenvolvimento	40 estudantes
Laboratório de Ensino 4	Hardware e Lógica digital	30 estudantes
Laboratório de Ensino 6	Programação e desenvolvimento	40 estudantes
Laboratório de Informática	Uso geral	40 estudantes
Espaço Maker	Aulas dinâmicas, com abordagem inovadora e prototipação	40 estudantes

Quadro 8.1: Laboratórios do Departamento de Computação voltados para o ensino da graduação.

8.2 Corpo docente e técnico

O Curso de Bacharelado em Ciência da Computação é atendido principalmente pelo Departamento de Computação (DC), que conta atualmente com 43 docentes (listados no Quadro 8.2) e 7 servidores lotados no DC (listados no Quadro 8.3).

Ao longo do período de quatro anos, os departamentos de Matemática, Estatística e Letras colaboram na formação do egresso com o oferecimento de disciplinas obrigatórias.

Nome	Titulação	Vínculo/Dedicação
Alan Demétrius Baria Valejo	Doutor	Efetivo/40h DE
Alexandre Luis Magalhães Levada	Doutor	Efetivo/40h DE
André Ricardo Backes	Doutor	Efetivo/40h DE
André Takeshi Endo	Doutor	Efetivo/40h DE
Auri Marcelo Rizzo Vincenzi	Doutor	Efetivo/40h DE
Cesar Henrique Comin	Doutor	Efetivo/40h DE
Daniel Lucrédio	Doutor	Efetivo/40h DE
Delano Medeiros Beder	Doutor	Efetivo/40h DE
Edilson Reis Rodrigues Kato	Doutor	Efetivo/40h DE
Ednaldo Brigante Pizzolato	Doutor	Efetivo/40h DE
Emerson Carlos Pedrino	Doutor	Efetivo/40h DE
Fabiano Cutigi Ferrari	Doutor	Efetivo/40h DE
Fredy João Valente	Doutor	Efetivo/40h DE
Helena de Medeiros Caseli	Doutor	Efetivo/40h DE
Helio Crestana Guardia	Doutor	Efetivo/40h DE
Heloisa de Arruda Camargo	Doutor	Efetivo/40h DE
Hermes Senger	Doutor	Efetivo/40h DE
Jander Moreira	Doutor	Efetivo/40h DE
Joice Lee Otsuka	Doutor	Efetivo/40h DE
Kelen Cristiane Teixeira Vivaldini	Doutor	Efetivo/40h DE
Luciano de Oliveira Neris	Doutor	Efetivo/40h DE
Marcela Xavier Ribeiro	Doutor	Efetivo/40h DE
Marcio Merino Fernandes	Doutor	Efetivo/40h DE
Marilde Terezinha Prado Santos	Doutor	Efetivo/40h DE
Mário César San Felice	Doutor	Efetivo/40h DE
Mauricio Fernandes Figueiredo	Doutor	Efetivo/40h DE
Murillo Rodrigo Petrucelli Homem	Doutor	Efetivo/40h DE
Murilo Coelho Naldi	Doutor	Efetivo/40h DE
Orides Morandin Junior	Doutor	Efetivo/40h DE
Paulo Matias	Doutor	Efetivo/40h DE
Pedro Henrique Bugatti	Doutor	Efetivo/40h DE
Priscila Tiemi Maeda Saito	Doutor	Efetivo/40h DE
Renato Bueno	Doutor	Efetivo/40h DE
Ricardo José Ferrari	Doutor	Efetivo/40h DE
Ricardo Menotti	Doutor	Efetivo/40h DE
Ricardo Rodrigues Ciferri	Doutor	Efetivo/40h DE
Roberto Ferrari Junior	Doutor	Efetivo/40h DE
Roberto dos Santos Inoue	Doutor	Efetivo/40h DE
Sandra Abib	Doutor	Efetivo/40h DE
Sergio Donizetti Zorzo	Doutor	Efetivo/40h DE
Valter Vieira de Camargo	Doutor	Efetivo/40h DE
Vânia Paula de Almeida Neris	Doutor	Efetivo/40h DE
Wanderley Lopes de Souza	Doutor	Efetivo/40h DE

Quadro 8.2: Corpo docente atuante no curso de Bacharelado em Ciência da Computação.

Nome	Atividade
Dalila Ariana de Abreu Bernardino	Assistente administrativo
Darli José Morcelli	Assistente administrativo
Ivan Rogério da Silva	Assistente administrativo
Jorgina Vera de Moraes	Servente de limpeza
Nicanor José Costa	Assistente administrativo
Paulo Cesar Donizeti Paris	Técnico de laboratório
Willian Câmara Corrêa	Técnico de laboratório

Quadro 8.3: Corpo técnico-administrativo atuante no DC.

Referências Bibliográficas

MINISTÉRIO DA EDUCAÇÃO - MEC. **Diretrizes Curriculares Nacionais de Cursos de Computação.** São Carlos, São Paulo, 16 de Novembro de 2016. Resolução nro 5.

UNIVERSIDADE FEDERAL DE SÃO CARLOS - UFSCAR. **Plano de Desenvolvimento Institucional.** São Carlos, São Paulo, 2024. P. 241.

UNIVERSIDADE FEDERAL DE SÃO CARLOS - UFSCAR. **Regimento Geral dos Cursos de Graduação.** São Carlos, São Paulo, Setembro de 2016. P. 115.

ZORZO, A. F. et al. **Referenciais de Formação para os Cursos de Graduação em Computação.** 2017. P. 153.

Apêndices

I. Regulamento do Estágio Curricular

Obrigatório e Não obrigatório

UNIVERSIDADE FEDERAL DE SÃO CARLOS
CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA

Coordenação do Curso de Bacharelado em Ciência da Computação

REGULAMENTO DO ESTÁGIO

TÍTULO I

Disposições Gerais

Art. 1 - Os estágios na UFSCar regem-se pela Lei nº 11.788/2008 e pelo Regimento Geral dos Cursos de Graduação da UFSCar. Conforme o regimento interno, os estágios podem ser obrigatórios ou não obrigatórios, em consonância com a legislação federal.

Art. 2 - O Estágio Obrigatório é um componente curricular que pode ser substituído pelo Trabalho de Conclusão de Curso para a integralização das horas em atividades curriculares do curso.

TÍTULO II

Objetivos

Art. 3 - O Estágio tem por objetivo fornecer ao estudante a experiência na área profissional para que ele possa colocar em prática os conhecimentos teóricos adquiridos no decorrer do curso, preparando-o para o exercício futuro da profissão. Para isso, a inserção na empresa é necessária possibilitando-lhe o contato com situações, problemas, processos reais, bem como com processos de tomada de decisão e realização de tarefas, complementando a sua formação teórica.

Art. 4 - O Estágio tem os seguintes objetivos específicos:

I - Consolidar o processo de formação do profissional bacharel em Ciência da Computação para o exercício da atividade profissional de forma integrada e autônoma.

II - Possibilitar oportunidades de interação dos estudantes com institutos de pesquisa, laboratórios e empresas que atuam nas diversas áreas da Ciência da Computação.

III - Desenvolver a integração Universidade-Comunidade, estreitando os laços de cooperação.

TÍTULO III

Caracterização do Estágio

Art. 5 - O Estágio deve ser desenvolvido nas áreas de conhecimento no âmbito da Ciência da Computação, mediante um Plano de Trabalho, elaborado em comum acordo entre as partes envolvidas.

Art. 6 - O Estágio em empresas estrangeiras é permitido desde que haja o registro de saída do estudante junto à SRInter e um Termo de Compromisso estabelecido com base na lei brasileira especificando, no mínimo: seguro, carga horária, supervisor na empresa e orientador na UFSCar.

Art. 7 - As atividades de Estágio poderão ser desenvolvidas durante as férias escolares ou durante o período letivo, embora a disciplina Estágio em Computação seja ofertada de acordo com os semestres letivos da UFSCar.

TÍTULO IV

Atividades que podem ser equiparadas ao Estágio

Art. 8 - Para este curso, de acordo com a Lei nº 11.788/2008 e sua alteração pela Lei nº 14.913/2024, as atividades que podem ser equiparadas ao Estágio são:

I - As atividades de extensão.

II - As iniciações científicas.

III - Os intercâmbios no exterior (p.ex. BEPE/FAPESP e BRAFITEC).

Art. 9 - A equiparação da atividade realizada com o Estágio deverá ser requisitada pelo estudante à Coordenação de Curso que fará a análise juntamente com a Coordenação de Estágio.

§1º - Apenas após a validação, por parte da Coordenação de Curso, é que as horas serão contabilizadas como Estágio.

§2º - A validação poderá ser de uma porcentagem das horas da atividade realizada e não necessariamente na totalidade delas.

Art. 10 - As atividades de que trata o Art. 8 poderão ser equiparadas ao Estágio desde que não tenham sido contabilizadas de outra forma para a integralização do curso (p.ex. como horas de atividades complementares ou de extensão).

TÍTULO V

Modalidades de Estágio

CAPÍTULO I

Estágio Obrigatório

Art. 11 - De acordo com a Lei 11.788/08 e Art. 33 do Regimento Geral dos Cursos de Graduação da UFSCar, Estágio Obrigatório é aquele cujo cumprimento da carga horária é requisito para aprovação e obtenção de diploma.

Art. 12 - O Estágio Obrigatório será desenvolvido obedecendo três etapas:

I - Planejamento: o qual se efetivará com a elaboração do plano de trabalho e formalização do termo de compromisso.

II - Supervisão e acompanhamento: que se concretizarão nos níveis profissional, didático-pedagógico e administrativo, desenvolvidos pelo supervisor local de estágio, pelo professor-orientador e pelo coordenador de estágio, respectivamente.

III - Avaliação: realizada nos níveis profissional e didático, desenvolvidos pelo supervisor local de estágio e professor-orientador, respectivamente.

Art. 13 - A carga horária mínima do Estágio Obrigatório é de 360 (trezentas e sessenta) horas a serem realizadas, preferencialmente, no 8º semestre do curso.

Art. 14 - O estudante só pode realizar o Estágio Obrigatório se cumprir as exigências especificadas no Capítulo III.

CAPÍTULO II

Estágio Não Obrigatório

Art. 15 - O Estágio Não Obrigatório é aquele desenvolvido como atividade opcional.

Art. 16 - O estudante só pode realizar o Estágio Não Obrigatório se cumprir as exigências especificadas no Capítulo IV.

Art. 17 - A jornada do Estágio Não Obrigatório deve ser compatível com as atividades acadêmicas.

Art. 18 - A carga horária desenvolvida no Estágio Não Obrigatório não poderá ser computada como carga horária de Estágio Obrigatório, sendo permitida sua contabilização como horas de atividades complementares se assim estiver previsto no PPC.

TÍTULO VI

Condições para Realização

CAPÍTULO III

Do Estágio Obrigatório

Art. 19 - O Estágio Obrigatório só pode ser realizado por estudante que satisfaça todas as condições a seguir:

I - Estar matriculado no curso de Bacharelado em Ciência da Computação.

II - Ter sido aprovado em, no mínimo, 1.830 horas em atividades curriculares da matriz curricular do curso.

III - Possuir um supervisor da parte concedente, para orientação, acompanhamento e avaliação do Estágio.

IV - Celebrar um Termo de Compromisso envolvendo o estudante, a parte concedente do Estágio e a UFSCar.

V - Elaborar um Plano de Atividades a serem desenvolvidas no Estágio, compatíveis com o projeto pedagógico do curso, o horário e o calendário escolar, de modo a contribuir para a efetiva formação profissional do estudante.

VI - Ser acompanhado efetivamente por um professor-orientador designado pela Coordenação de Estágio e por um supervisor da parte concedente, sendo ambos responsáveis por examinar e aprovar os relatórios periódicos e final,

elaborados pelo estagiário.

Art. 20 - O requisito mínimo de horas em atividades curriculares concluídas pelo estudante de que trata este capítulo refere-se apenas a disciplinas obrigatórias e optativas da matriz curricular do curso e não outras que o estudante tenha feito, como eletivas, outras optativas e ACIEPEs.

CAPÍTULO IV

Do Estágio Não Obrigatório

Art. 21 - O Estágio Não Obrigatório só poderá ser realizado por estudante que satisfaça todas as condições a seguir:

I - Estar matriculado no curso de Bacharelado em Ciência da Computação.

II - Ter sido aprovado em, no mínimo, 1.530 horas em atividades curriculares da matriz curricular do curso.

III - Possuir um supervisor da parte concedente para orientação e acompanhamento do Estágio.

IV - Celebrar um Termo de Compromisso envolvendo o estudante, a parte concedente do Estágio e a UFSCar.

V - Elaborar um Plano de Atividades a serem desenvolvidas no Estágio, compatíveis com o projeto pedagógico do curso, o horário e o calendário escolar, de modo a contribuir para a efetiva formação profissional do estudante.

VI - Ser acompanhado efetivamente por um professor-orientador designado pela Coordenação de Estágio e por um supervisor da parte concedente, sendo ambos responsáveis por examinar e aprovar os relatórios periódicos e final, elaborados pelo estagiário.

Art. 22 - O requisito mínimo de horas em atividades curriculares concluídas pelo estudante de que trata este capítulo refere-se apenas a disciplinas obrigatórias e optativas da matriz curricular do curso e não outras que o estudante tenha feito, como eletivas, outras optativas e ACIEPEs.

TÍTULO VII

Jornada de Atividade em Estágio

Art. 23 - De acordo com a Lei 11.788/08, a jornada de atividade em Estágio será definida de comum acordo entre a Instituição de Ensino, a parte concedente e o estudante estagiário, devendo constar do Termo de Compromisso e ser compatível com as atividades curriculares do semestre.

§1º - A compatibilidade entre as atividades de Estágio e as atividades curriculares é satisfeita quando somando-se a carga horária semanal prevista para o Estágio e para as atividades curriculares que o estudante fará no semestre não ultrapassem o total de 40 (quarenta) horas, ficando limitado a jornada de estágio a no máximo 30 (trinta) horas semanais.

§2º - O estudante poderá fazer 40 (quarenta) horas de estágio semanais desde que ele tenha integralizado as horas em atividades curriculares exigidas em sua matriz curricular e desde que a única dependência de sua formação seja o Estágio.

TÍTULO VIII

Coordenação de Estágio

Art. 24 - A Coordenação de Estágio do Curso de Bacharelado em Ciência da Computação será realizada por um docente do Departamento de Computação responsável pela disciplina Estágio em Computação.

Parágrafo único - A Coordenação de Estágio contará com o apoio de servidor técnico-administrativo lotado na Coordenação do Curso de Bacharelado em Ciência da Computação ou no Departamento de Computação, para o desempenho de suas atividades.

Art. 25 - As atribuições da Coordenação de Estágio são:

I - Estar em contato com empresas interessadas em contratar estagiários.

II - Informar os estudantes sobre as regras brasileiras e institucionais do Estágio.

III - Orientar os estudantes quanto ao preenchimento correto do Termo de Compromisso de Estágio.

IV - Avaliar o Plano de Trabalho de Estágio.

V - Designar Orientador do Estágio Curricular.

VI - Coordenar a tramitação de todos os instrumentos jurídicos: termos de compromisso, requerimentos, cartas de apresentação, cartas de autorização ou outros documentos necessários para que o Estágio seja oficializado, bem como a guarda destes.

VII - Coordenar as atividades de avaliações do Estágio Obrigatório.

Parágrafo único - Quando houver necessidade de celebração de convênio institucional para a habilitação de oportunidades de estágio, a Coordenação de Estágios encaminhará a demanda à Coordenadoria de Estágio e Mobilidade da Pró-Reitoria de Graduação, para celebração da parceria.

TÍTULO IX

Orientação e Supervisão do Estágio

CAPÍTULO V

Orientação do Estágio

Art. 26 - O professor responsável pela orientação do estudante durante o Estágio será um professor do curso de Ciência da Computação denominado de professor-orientador.

Art. 27 - O professor-orientador será responsável pelo acompanhamento e avaliação das atividades do estagiário e terá as seguintes atribuições:

I - Orientar o estudante na condução de seu Plano de Trabalho de Estágio e na elaboração dos relatórios de Estágio.

II - Orientar o estagiário quanto aos aspectos técnicos, científicos e éticos das atividades de Estágio.

III - Supervisionar o desenvolvimento do Plano de Trabalho de Estágio mantendo sempre que possível contato com o supervisor local do Estágio.

IV - Estabelecer datas para reuniões de acompanhamento com o estagiário e para a entrega de relatórios das atividades realizadas na empresa.

V - Avaliar o Estágio, especialmente os relatórios, e encaminhar ao Coordenador de Estágio seu parecer.

CAPÍTULO VI

Supervisão do Estágio

Art. 28 - O Supervisor do estagiário deverá ser um profissional que atue no local no qual o estudante desenvolverá suas atividades de Estágio e terá as seguintes atribuições:

I - Garantir o acompanhamento contínuo e sistemático do estagiário, desenvolvendo a sua orientação e assessoramento dentro do local de estágio. Não é necessário que o supervisor seja Cientista da Computação, mas deve ser um profissional que tenha extensa experiência na área de atuação.

II - Informar o professor-orientador ou a Coordenação de Estágio as ocorrências relativas ao estagiário, buscando assim estabelecer um intercâmbio permanente entre a Universidade e a Empresa.

III - Apresentar um relatório de avaliação do estagiário à Coordenação de Estágio.

TÍTULO X

Obrigações do estagiário

Art. 29 - O estagiário, durante o desenvolvimento das atividades de Estágio, terá as seguintes obrigações:

I - Apresentar documentos exigidos pela UFSCar e pela concedente.

II - Seguir as determinações do Termo de Compromisso de Estágio.

III - Cumprir integralmente o horário estabelecido pela concedente, observando assiduidade e pontualidade.

IV - Manter sigilo sobre conteúdo de documentos e de informações confidenciais referentes ao local de Estágio.

V - Acatar orientações e decisões do supervisor local de Estágio, quanto às normas internas da concedente.

VI - Efetuar registro de sua frequência no Estágio.

VII - Elaborar e entregar relatório das atividades de Estágio e outros documentos nas datas estabelecidas pelo professor-orientador ou Coordenação de Estágio. O relatório final deve ser elaborado em formato acadêmico, contemplando a relação entre as atividades desenvolvidas e o conhecimento adquirido pelo estagiário na área.

VIII - Respeitar as orientações e sugestões do supervisor local de Estágio.

IX - Manter contato com o professor-orientador sempre que julgar necessário.

X - Assumir o Estágio com responsabilidade, zelando pelo bom nome da Instituição Concedente e do curso de Ciência da Computação.

XI - Matricular-se na disciplina Estágio em Computação no semestre letivo no qual estiver realizando o Estágio.

TÍTULO XI

Formalização do Termo de Compromisso de Estágio

Art. 30 - O Termo de Compromisso de Estágio deverá ser celebrado entre o estudante, a parte concedente do Estágio e a UFSCar, estabelecendo:

I - O Plano de Atividades a serem realizadas, que figurará em anexo ao respectivo Termo de Compromisso.

II - As condições de realização do Estágio, em especial, a duração e a jornada de atividades, respeitada a legislação vigente e as regras estabelecidas neste regulamento.

III - As obrigações do estagiário, da concedente e da UFSCar.

IV - O valor da bolsa ou outra forma de contraprestação devida ao estagiário, e o auxílio-transporte, a cargo da concedente, quando for o caso.

V - O direito do estagiário ao recesso das atividades na forma da legislação vigente.

VI - O seguro contra acidente pessoal, assegurado pela concedente em favor do estagiário, que figurará em anexo ao respectivo Termo de Compromisso após sua realização.

TÍTULO XII

Disposições Finais

Art. 31 - O presente Regulamento de Estágio entra em vigor a partir da data de sua aprovação pelo Conselho de Coordenação de Curso.

Art. 32 - Os casos omissos serão resolvidos pela Coordenação de Curso em conjunto com a Coordenação de Estágio, cabendo ao Conselho do Curso analisar a necessidade de atualização deste Regulamento.

II. Regulamento do Trabalho de Conclusão de Curso

UNIVERSIDADE FEDERAL DE SÃO CARLOS
CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA

Coordenação do Curso de Bacharelado em Ciência da Computação

REGULAMENTO DO TRABALHO DE CONCLUSÃO DE CURSO

TÍTULO I

Disposições Gerais

Art. 1 - O Trabalho de Conclusão de Curso (TCC) está estruturado pelo Regimento Geral dos Cursos de Graduação da UFSCar.

Art. 2 - No Bacharelado em Ciência da Computação, o TCC é um componente curricular opcional e constitui-se em um trabalho acadêmico de produção orientada, que sintetiza e integra conhecimentos, competências e habilidades adquiridos durante o curso.

Parágrafo único - O TCC pode substituir o Estágio Obrigatório.

TÍTULO II

Objetivos

Art. 3 - O TCC tem por objetivo propiciar ao estudante a oportunidade de reflexão, análise e crítica, articulando a teoria e a prática, resguardando o nível adequado de autonomia intelectual do estudante.

Parágrafo único - O TCC deve versar sobre qualquer área do conhecimento da Ciência da Computação.

TÍTULO III

Condições para Realização

Art. 4 - O TCC só pode ser realizado por estudante que satisfaça todas as condições a seguir:

I - Estar matriculado no curso de Bacharelado em Ciência da Computação.

II - Ter sido aprovado em, no mínimo, 2.310 horas em atividades curriculares da matriz curricular do curso.

III - Ser orientado por um docente da UFSCar, com titulação de doutor e reconhecida experiência acadêmica, aqui denominado professor-orientador.

IV - Estar matriculado na disciplina Trabalho de Conclusão de Curso no semestre letivo de realização do TCC.

Art. 5 - O requisito mínimo de horas em atividades curriculares concluídas pelo estudante de que trata este capítulo refere-se apenas a disciplinas obrigatórias e optativas da matriz curricular do curso e não outras que o estudante tenha feito, como eletivas, outras optativas e ACIEPEs.

TÍTULO IV

Desenvolvimento do TCC

Art. 6 - No curso de Ciência da Computação, o TCC é realizado no 8º (oitavo) semestre em caráter optativo com o Estágio Obrigatório, onde a disciplina é nomeada como Trabalho de Conclusão do Curso.

Art. 7 - O TCC é um trabalho acadêmico de caráter de pesquisa ou técnico, dessa forma poderá ser substituído por um trabalho de Iniciação Científica (IC).

Parágrafo único - O estudante que tenha realizado uma IC poderá solicitar à Coordenação do Curso a equivalência da IC com o TCC conforme especificado no Capítulo III.

CAPÍTULO I

A disciplina de Trabalho de Conclusão de Curso

Art. 8 - A integralização da carga horária exigida para a realização do TCC se concretizará mediante a frequência e aprovação na disciplina Trabalho de Conclusão de Curso.

Parágrafo único - Caberá ao docente encarregado dessa disciplina estabelecer cronogramas e critérios de avaliação condicionados à aprovação da banca.

CAPÍTULO II

Orientação do TCC

Art. 9 - O TCC deverá ser desenvolvido mediante a orientação de um docente da UFSCar, com titulação de doutor e reconhecida experiência acadêmica, aqui denominado professor-orientador.

Parágrafo único - Será permitida a co-orientação com a participação de profissionais externos à UFSCar.

Art. 10 - O professor-orientador é escolhido pelo estudante de acordo com a maior proximidade do tema a ser desenvolvido, ou seja, deve ser um docente que possua a expertise do tema na forma de sua concepção e modelagem e que tenha conhecimento das técnicas para fazê-lo.

Art. 11 - O professor-orientador deverá acompanhar o desenvolvimento do trabalho durante todo o seu período de realização, orientando constantemente o estudante em sua execução.

Art. 12 - Além do professor-orientador, haverá também um docente responsável pela disciplina Trabalho de Conclusão de Curso conforme detalhado no Capítulo I.

CAPÍTULO III

Equivalência do TCC com uma IC

Art. 13 - O estudante que tenha realizado uma IC poderá solicitar à Coordenação do Curso a equivalência da IC com o TCC.

Art. 14 - O procedimento para atestar a equivalência da IC com o TCC seguirá duas etapas:

I - Etapa 1: O estudante solicita a análise da equivalência da IC comprovando sua conclusão com êxito por meio da apresentação do relatório final à Coordenação do Curso. A Coordenação do Curso designa uma comissão de professores para analisar se a IC realizada equivale ao TCC. Caso a equivalência da IC com o TCC seja aprovada, o estudante será encaminhado para a segunda etapa.

II - Etapa 2: Caso a equivalência da IC com o TCC seja aprovada, o estudante deverá apresentar escrita e oralmente o trabalho para uma banca examinadora. O documento a ser entregue para avaliação da banca pode ser o próprio relatório da IC, desde que este siga os modelos PIBIC ou FAPESP. As regras de formação dessa banca examinadora e a avaliação são as mesmas do Capítulo IV.

Art. 15 - A IC utilizada para substituir o TCC não poderá ser computada como horas em outro tipo de atividade curricular, como horas de atividade complementar por exemplo.

TÍTULO V

Avaliação

Art. 16 - O TCC será avaliado por uma banca examinadora formada como especificado no Capítulo IV.

§1º - O trabalho final deverá ser apresentado em forma de monografia e realizada uma exposição oral a membros de uma banca de avaliação.

§2º - A monografia deverá seguir o rigor acadêmico de autenticidade (caso contrário é considerado plágio), o formalismo e os critérios de qualidade, de acordo com as normas atuais.

§3º - No texto escrito serão avaliadas a redação, a qualidade do trabalho realizado e as contribuições para a formação

do estudante, bem como sua adequação às normas da Associação Brasileira de Normas Técnicas (ABNT).

§4º - Na apresentação oral serão avaliadas a exposição do trabalho realizado e a arguição pelos examinadores.

§5º - A apresentação da monografia deverá ser realizada em sessão pública dentro das datas estabelecidas previamente no início de cada semestre.

§6º - A defesa de TCC deve ocorrer preferencialmente de forma presencial, havendo a possibilidade, a critério do professor-orientador, de ter formato remoto ou híbrido.

CAPÍTULO IV

Composição da Banca

Art. 17 - A banca será composta por um mínimo de três integrantes e um máximo de quatro, sendo pelo menos dois professores da UFSCar.

§1º - O professor-orientador é membro natural da banca examinadora e irá presidir a sessão.

§2º - Para os demais membros da banca, são elegíveis todas as pessoas com nível superior completo.

Art. 18 - A indicação de nomes de membro da banca, bem como a definição da data e reserva de sala é de responsabilidade do professor-orientador e do estudante, respeitando o cronograma pré-estabelecido pelo docente responsável pela disciplina ou Coordenação do Curso.

TÍTULO VI

Disposições Finais

Art. 19 - O presente Regulamento do Trabalho de Conclusão de Curso entra em vigor a partir da data de sua aprovação pelo Conselho de Coordenação de Curso.

Art. 20 - Os casos omissos serão resolvidos pelo docente responsável pela disciplina de Trabalho de Conclusão de Curso, e em última instância pela Coordenação do Curso.

III. Regulamento de Atividades Curriculares de Extensão

UNIVERSIDADE FEDERAL DE SÃO CARLOS
CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA

Coordenação do Curso de Bacharelado em Ciência da Computação

REGULAMENTO DE ATIVIDADES CURRICULARES DE EXTENSÃO

TÍTULO I

Disposições Gerais

Art. 1 - As Atividades Curriculares de Extensão (ACES) são atividades de caráter extensionista realizadas pelo estudante ao longo de seu curso de graduação.

Parágrafo único - Este Regulamento, definido com base na Resolução CoG 02/2023, estabelece diretrizes para as ACES consideradas na integralização curricular, de acordo com os objetivos do curso.

TÍTULO II

Objetivos

Art. 2 - As ACES, como parte da Extensão Universitária, tem como objetivo promover a interação transformadora entre as instituições de ensino superior e os outros setores da sociedade, por meio da produção, aplicação e divulgação do conhecimento, em articulação permanente com o ensino e a pesquisa.

Parágrafo único - A Extensão Universitária constitui-se em processo interdisciplinar, político educacional, cultural, científico e tecnológico.

TÍTULO III

Tipos de ACEs

Art. 3 - As ACEs consideradas para integralização curricular neste curso são dos seguintes tipos:

I - Atividades Curriculares Obrigatórias, Optativas ou Eletivas com carga horária integral ou parcial voltada à abordagem extensionista (ACEs do tipo I).

II - Atividades Complementares de Extensão: Ações de extensão, com ou sem bolsa, com aprovação registrada na Pró-Reitoria de Extensão nas modalidades de projetos, cursos, oficinas, eventos, prestação de serviços e ACIEPEs não previstas na matriz curricular (ACEs do tipo III).

Art. 4 - Atividades derivadas de iniciativas da UFSCar como Empresas Juniores, Cursinhos Pré-Vestibulares, Programa de Educação Tutorial (PET) e Programa Institucional de Bolsas de Iniciação à Docência (PIBID) poderão ser consideradas ACEs do tipo III, desde que estejam registradas como ações de extensão na Pró-Reitoria de Extensão da UFSCar.

Art. 5 - As ACEs do tipo III devem indicar a categoria de participação do estudante como membro da equipe de trabalho da ACE ou público-alvo (participante inscrito).

§1º - No caso das ACIEPEs, pela natureza da sua concepção, todos os inscritos têm participação categorizada de forma equivalente à da equipe de trabalho.

§2º - Só serão contabilizadas horas em ACEs do tipo III para estudantes registrados na equipe de trabalho da atividade de extensão.

Art. 6 - Os estágios obrigatório e não obrigatório seguem normativas próprias e não podem ser considerados como ACE.

TÍTULO IV

Carga Horária Exigida

Art. 7 - O estudante deve cumprir no mínimo 360 (trezentas) horas em atividades curriculares de extensão para concluir o curso.

Art. 8 - Para as ACEs do tipo I, a carga horária destinada à extensão universitária deverá ser indicada na ficha de caracterização da atividade.

Parágrafo único - O cômputo e o registro da carga horária das ACEs do tipo I ocorrerá automaticamente no SIGA, com a inscrição e aprovação do estudante na respectiva atividade curricular.

Art. 9 - O registro da carga horária das ACEs do tipo III será realizado pela Coordenação do Curso diretamente no SIGA com base no documento oficial relatando a conclusão das ações de extensão.

Parágrafo único - A contabilização da carga horária de extensão para ACEs do tipo III deverá ser requisitada pelo estudante à Coordenação de Curso assim que o documento de conclusão da atividade for obtido.

Art. 10 - O estudante que já tenha integralizado a carga horária mínima especificada no Art. 7 e realize novas atividades que tenham carga horária extensionista, como ACIEPEs e PIEs, poderá requisitar que as mesmas sejam contabilizadas como horas de atividades complementares.

TÍTULO V

Integralização da Carga Horária

Art. 11 - A integralização da carga horária estabelecida no Art. 7 ocorrerá por meio da contabilização de horas com a conclusão das seguintes atividades:

I - Atividade Curricular (disciplina) obrigatória Computação e Sociedade que contabiliza carga horária integral extensionista de 60 horas, bem como outras disciplinas eletivas com carga horária de extensão (ACEs do tipo I).

II - ACIEPEs que contabilizam carga horária integral extensionista de 60 horas cada (ACE do tipo III).

III - Ações de extensão, com ou sem bolsa, registrada na ProEx que contabilizam carga horária integral extensionista conforme Quadro 1 ao final deste documento (ACE do tipo III).

Art. 12 - O número mínimo e máximo de horas para cada um dos três tipos de ACEs é:

I - ACE do tipo I: mínimo 60 e máximo 240.

II - ACE do tipo II: mínimo 0 e máximo 0. O BCC não prevê ACEs do tipo II.

III - ACE do tipo III: mínimo 120 e máximo 300.

Art. 13 - Atividades não previstas neste Regulamento serão analisadas pelo Conselho de Curso que poderá indicar a necessidade de sua atualização.

TÍTULO VI

Caracterização de Disciplina com Caráter Extensionista (ACE do tipo I)

Art. 14 - De acordo com a Instrução Normativa ProGrad nº 2 de 20 de dezembro de 2024 (Art. 2), para que sejam reconhecidas como ACEs, as atividades deverão atender aos seguintes princípios:

I - Contribuição para a formação integral do estudante estimulando sua formação como cidadão crítico e responsável.

II - Estabelecimento de diálogo construtivo e transformador com os demais setores da sociedade brasileira e/ou internacional.

III - Envolvimento proativo dos estudantes na promoção de iniciativas que expressam o compromisso social das instituições de ensino superior com todas as áreas e prioritariamente as de comunicação, cultura, direitos humanos e justiça, educação, meio ambiente, saúde, tecnologia e produção, trabalho, em consonância com as políticas ligadas às diretrizes para a educação ambiental, educação linguística, educação das relações étnico-raciais, direitos humanos e educação indígena, considerando a interprofissionalidade e a interdisciplinaridade.

IV - Contribuição ao enfrentamento de questões no contexto local, regional, nacional ou internacional, respeitando-se os objetivos de desenvolvimento sustentável (ODS) definidos pela ONU.

Parágrafo único - Os temas explorados nessas atividades devem se encaixar nas ODS da ONU.

Art. 15 - De acordo com a Instrução Normativa ProGrad nº 2 de 20 de dezembro de 2024 (Art. 5), os Planos de Ensino das ACEs dos tipos I e II, deverão conter a descrição da característica extensionista nos itens: objetivos específicos, estratégias de ensino, atividades dos estudantes, recursos didático-tecnológicos a serem utilizados,

procedimentos de avaliação dos estudantes e bibliografias.

Art. 16 - A disciplina candidata a ser considerada uma ACE deve apresentar ao menos uma das seguintes características:

I - Projeto e/ou implementação de hardware e/ou software real.

II - Disponibilização em repositório de software público.

III - Interação direta/indireta com público externo ao curso.

IV - Disponibilização de software para teste público.

Art. 17 - A disciplina extensionista (ACE do tipo I) tem a obrigatoriedade do conteúdo produzido ser externado à comunidade, o que pode ser realizado na forma de *workshops*, palestras, apresentação de pôsteres, redes sociais, fóruns de discussão, vídeos publicados em caráter público na *Web*, webconferência *on line*, salas virtuais (como Google Classroom e Moodle), repositórios públicos de software entre outros, garantindo assim a interação com o público externo à disciplina (comunidade interna e externa à UFSCar).

Art. 18 - O conteúdo da disciplina extensionista ACE do tipo I a ser externado ao público interno e externo à UFSCar pode ser parcial, durante a execução da disciplina, ou total, incluindo a apresentação de projetos, que podem ser divulgados prévia e amplamente nos meios de comunicação digital, inclusive nos oficiais, em pelo menos um deles, como o *site* do Departamento de Computação (DC) da UFSCar, o Boletim informativo da UFSCar, o Boletim de oportunidades da UFSCar, entre outros.

TÍTULO VII

Disposições Finais

Art. 19 - O presente Regulamento de Atividades Curriculares de Extensão entra em vigor a partir da data de sua aprovação pelo Conselho de Coordenação de Curso.

Art. 20 - Os casos omissos serão resolvidos pela Coordenação de Curso, cabendo ao Conselho do Curso estipular a carga horária a ser considerada em cada um dos casos e analisar a necessidade de atualização deste Regulamento.

Nome da Atividade	Cômputo em Horas de Atividades Complementares (HAC)	Tipo de comprovação exigida (Todas essas atividades devem estar registradas na ProEx)	Limitações
Participação no PET (bolsista ou não-bolsista) (Veja Art. 4)	165 (cento e sessenta e cinco) HAC por ano ou proporcional	Certificado de participação no PET emitido pelo órgão competente dentro da universidade	Mínimo de 6 (seis) meses
Participação em Projeto Integrador Extensionista (PIE)	210 (duzentas e dez) HAC por ano	Declaração de aprovação da banca final	–
Participação em Empresa Jr.	110 (cento e dez) HAC por ano ou proporcional	Declaração do professor coordenador	–
Organização da SECOMP	120 (cento e vinte) HAC para coordenadores gerais, limitado a 3 (três) coordenadores por edição 120 (cento e vinte) HAC para diretores, limitado a 1 (um) diretor por diretoria por edição 90 (noventa) HAC para membros	Declaração do professor coordenador da SECOMP	Máximo de 2 (duas) participações em comissão organizadora da SECOMP
Participação em Atividades de Extensão registrada na ProEx, na área do curso (bolsista ou voluntário) Exemplo: Maratona de programação	13,75 (treze vírgula setenta e cinco) HAC por mês de vigência da atividade e considerando carga horária de 12 (doze) horas semanais ou proporcional	Certificado de participação emitido pelo coordenador ou órgão competente dentro da universidade	Máximo de 13,75 (treze vírgula setenta e cinco) HAC por mês
Participação em Atividade de Extensão registrada na ProEx, fora da área do curso (bolsista ou voluntário)	1 (uma) HAC por hora de atividade	Certificado de participação emitido pelo coordenador ou órgão competente dentro da universidade	Máximo de 60 (sessenta) HAC
Participação na organização de eventos na área do curso	1,5 (um vírgula cinco) HAC por hora	Certificado de participação como organizador constando o número de horas do evento	Máximo de 82,5 (oitenta e duas vírgula cinco) HAC

OBS.: Devido a limitações do sistema, no momento de lançamento de horas complementares será realizado o arredondamento para o menor inteiro superior.

Casos omissos e de cômputo parcial de horas complementares serão apreciados pelo Conselho de Curso.

Quadro 1: Atividades Complementares de Extensão

IV. Regulamento de Atividades Complementares

UNIVERSIDADE FEDERAL DE SÃO CARLOS
CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA

Coordenação do Curso de Bacharelado em Ciência da Computação

REGULAMENTO DE ATIVIDADES COMPLEMENTARES

TÍTULO I

Disposições Gerais

Art. 1 - As Atividades Complementares (ACs) são todas e quaisquer atividades de caráter acadêmico, científico e cultural realizadas pelo estudante ao longo de seu curso de graduação.

Parágrafo único - Este Regulamento estabelece uma relação de atividades complementares a serem consideradas para fim de integralização curricular, de acordo com os objetivos do curso.

TÍTULO II

Objetivos

Art. 2 - As ACs, no contexto do curso de Ciência da Computação, tem como objetivo complementar a formação do profissional de excelência, dotando-o de habilidades e competências complementares para sua atuação profissional.

Parágrafo único - As ACs são todas aquelas que expõem o estudante a experiências inovadoras e enriquecedoras durante seu período acadêmico, complementando a formação prevista na matriz curricular.

TÍTULO III

Carga Horária Exigida

Art. 3 - O estudante deve cumprir no mínimo 210 (duzentas e dez) horas em atividades complementares para concluir o curso.

Art. 4 - Nos Quadros 1 e 2 são apresentadas as atividades complementares possíveis de serem realizadas bem como as respectivas cargas horárias e documentos de comprovação a serem apresentados.

§1º - A contabilização da carga horária em atividades complementares (HAC) deverá ser requisitada pelo estudante à Coordenação de Curso assim que o documento de comprovação da realização da atividade for obtido.

§2º - Atividades que tenham carga horária extensionista (Quadro 2) só poderão ser contabilizadas como horas de atividades complementares se a carga horária de extensão já tiver sido integralizada, caso contrário serão contabilizadas como horas em atividades extensionistas.

TÍTULO IV

Integralização da Carga Horária

Art. 5 - A integralização da carga horária exigida no Art. 3 ocorrerá por meio da contabilização de horas com a conclusão das atividades previstas nos Quadros 1 e 2.

TÍTULO V

Disposições Finais

Art. 6 - O presente Regulamento de Atividades Complementares entra em vigor a partir da data de sua aprovação pelo Conselho de Coordenação de Curso.

Art. 7 - Os casos omissos serão resolvidos pela Coordenação de Curso, cabendo ao Conselho do Curso estipular a carga horária a ser considerada em cada um dos casos e analisar a necessidade de atualização deste Regulamento.

Nome da Atividade	Cômputo em Horas de Atividades Complementares (HAC)	Tipo de comprovação exigida	Observação	Limitações
Iniciação Científica e suas variações (PIBIC, PIBIT, ICT-SR, PIBIC-Af, FAPESP) (bolsista ou voluntário)	165 (cento e sessenta e cinco) HAC por ano ou proporcional	Certificado de conclusão da IC emitido pelo órgão competente dentro da universidade (ex: CoPICT) ou agência de fomento	São aceitos certificados emitidos também por outras universidades.	Mínimo de 6 (seis) meses
Atuação como monitor em disciplinas de graduação da UFSCar (bolsista ou voluntário)	82,5 (oitenta e duas vírgula cinco) HAC por monitoria	Certificado de atuação emitido pelo órgão competente dentro da UFSCar	–	Máximo de 2 (duas) monitorias
Participação em intercâmbios internacionais na área do curso Exemplo: BEPE FAPESP, estágios no exterior, visitas de curta duração etc.	13,75 (treze vírgula setenta e cinco) HAC por mês de intercâmbio	Documento emitido pelo órgão competente dentro da universidade ou agência de fomento	–	–
Representação em órgãos colegiados , como conselhos de curso, departamentais e conselhos superiores da universidade	60 (sessenta) HAC por mandato de 1 (um) ano para membro titular 30 (trinta) HAC por mandato de 1 (um) ano para membro suplente	Documento comprobatório (ex: ata da comissão eleitoral ou portaria que montou o conselho)	Contabilizado após a finalização do mandato	Máximo de 82,5 (oitenta e duas vírgula cinco) HAC
Participação como ouvinte em seminário, palestra, mesa-redonda, minicurso ou tutorial	1 (uma) HAC por hora do evento	Certificado de participação constando o número de horas do evento	–	Máximo de 60 (sessenta) HAC
Participação como ouvinte em congresso (científico ou tecnológico)	8 (oito) HAC por dia de evento	Certificado de participação emitido pela organização do evento	Desde que não tenha sido contabilizado no item anterior	Máximo de 60 (sessenta) HAC
Registro de Software no INPI (Instituto Nacional de Propriedade Intelectual)	82,5 (oitenta e duas vírgula cinco) HAC por registro	Certificado do registro	–	Máximo de 2 (dois) registros
Participação como ministrante de minicurso ou tutorial na área do curso ou apresentação de trabalho em congresso (científico ou tecnológico)	2 (duas) HAC por hora do minicurso ou apresentação	Certificado de participação como apresentador ou ministrante (constando o número de horas do evento)	–	Máximo de 82,5 (oitenta e duas vírgula cinco) HAC
Participação em Centro Acadêmico	80 (oitenta) HAC por mandato de 1 ano ou proporcional, para cada um dos membros das diretorias que tiverem frequência mínima de 75% nas reuniões semanais (com a possibilidade de até 3 faltas justificadas)	Atas das reuniões	O pedido deverá ser apresentado à secretaria da coordenação de curso, acompanhado das atas das reuniões.	Mínimo de 6 (seis) meses
Disciplinas eletivas (qualquer disciplina ofertada na UFSCar)	Carga horária da disciplina	Aprovação da disciplina no Histórico escolar	–	Máximo de 3 (três) eletivas
OBS.: Devido a limitações do sistema, no momento de lançamento de horas complementares será realizado o arredondamento para o menor inteiro superior.				
Casos omissos e de cômputo parcial de horas complementares serão apreciados pelo Conselho de Curso.				

Quadro 1: Atividades Complementares

Nome da Atividade	Cômputo em Horas de Atividades Complementares (HAC)	Tipo de comprovação exigida (Todas essas atividades devem estar registradas na ProEx)	Limitações
Participação no PET (bolsista ou não-bolsista) (Veja Art. ??)	165 (cento e sessenta e cinco) HAC por ano ou proporcional	Certificado de participação no PET emitido pelo órgão competente dentro da universidade	Mínimo de 6 (seis) meses
Participação em Projeto Integrador Extensionista (PIE)	210 (duzentas e dez) HAC por ano	Declaração de aprovação da banca final	–
Participação em Empresa Jr.	110 (cento e dez) HAC por ano ou proporcional	Declaração do professor coordenador	–
Organização da SECOMP	120 (cento e vinte) HAC para coordenadores gerais, limitado a 3 (três) coordenadores por edição 120 (cento e vinte) HAC para diretores, limitado a 1 (um) diretor por diretoria por edição 90 (noventa) HAC para membros	Declaração do professor coordenador da SECOMP	Máximo de 2 (duas) participações em comissão organizadora da SECOMP
Participação em Atividades de Extensão registrada na ProEx, na área do curso (bolsista ou voluntário) Exemplo: Maratona de programação	13,75 (treze vírgula setenta e cinco) HAC por mês de vigência da atividade e considerando carga horária de 12 (doze) horas semanais ou proporcional	Certificado de participação emitido pelo coordenador ou órgão competente dentro da universidade	Máximo de 13,75 (treze vírgula setenta e cinco) HAC por mês
Participação em Atividade de Extensão registrada na ProEx, fora da área do curso (bolsista ou voluntário)	1 (uma) HAC por hora de atividade	Certificado de participação emitido pelo coordenador ou órgão competente dentro da universidade	Máximo de 60 (sessenta) HAC
Participação na organização de eventos na área do curso	1,5 (um vírgula cinco) HAC por hora	Certificado de participação como organizador constando o número de horas do evento	Máximo de 82,5 (oitenta e duas vírgula cinco) HAC

OBS.: Devido a limitações do sistema, no momento de lançamento de horas complementares será realizado o arredondamento para o menor inteiro superior.

Casos omissos e de cômputo parcial de horas complementares serão apreciados pelo Conselho de Curso.

Quadro 2: Atividades Complementares de Extensão

V. Regulamento do Projeto Integrador Extensionista

UNIVERSIDADE FEDERAL DE SÃO CARLOS
CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA

Coordenação do Curso de Bacharelado em Ciência da Computação

REGULAMENTO DO PROJETO INTEGRADOR EXTENSIONISTA

TÍTULO I

Disposições Gerais

Art. 1 - O Projeto Integrador Extensionista (PIE) é uma atividade de extensão desenvolvida no Departamento de Computação que visa atender a Lei Nº 13.005 de 25 de junho de 2014, a qual aprova o Plano Nacional de Educação (PNE) para o período de 2014 a 2024 e afirma que devem-se intensificar as atividades de extensão nos cursos de graduação, sendo recomendado que 10% da carga horária do curso seja destinada a atividades de extensão.

Art. 2 - Este Regulamento estabelece as diretrizes para a realização do PIE de acordo com os objetivos do curso.

TÍTULO II

Objetivos

Art. 3 - O objetivo do PIE é propiciar aos estudantes um embasamento prático dos conceitos teóricos adquiridos por meio dos conteúdos programáticos ministrados em sala de aula. Tais projetos devem, obrigatoriamente, atender demandas externas ao departamento, inclusive demandas de empresas, caracterizando-se como projetos de extensão, sendo supervisionados por um professor-orientador da Universidade.

Art. 4 - A intenção é que o PIE aproxime-se da forma como os estudantes atuarão na vida profissional: agindo positivamente na solução de problemas técnicos, sociais, políticos e econômicos, objetivando o desenvolvimento socioeconômico nas perspectivas local, regional, nacional e/ou internacional.

Art. 5 - Os objetivos específicos do PIE são:

- I - Propiciar aos estudantes identificar com mais clareza a relação existente entre as disciplinas cursadas, além de promover cada vez mais a interação dos conteúdos apresentados;
- II - Propiciar aos estudantes compreender quais conhecimentos e tecnologias podem ser combinadas e adequadas para a resolução de cada problema;
- III - Possibilitar aos estudantes fundamentos e aspectos metodológicos iniciais para realização de trabalhos profissionais, estimulando o espírito cooperativo e sensibilizando-os para a importância do trabalho em equipe;
- IV - Incentivar aos estudantes na identificação de problemas que afetem a comunidade externa ao DC e que possam ser resolvidos por meio do uso de técnicas computacionais;
- V - Possibilitar aos estudantes a troca de experiências e o desenvolvimento da capacidade de organização para o desenvolvimento de trabalho em equipe;
- VI - Incentivar aos estudantes a busca por inovação e o registro de propriedade intelectual e/ou patente no Instituto Nacional de Proteção Intelectual (INPI), com apoio da Agência de Inovação da Fundação de Apoio Institucional (FAI) da UFSCar;
- VII - Propiciar aos estudantes o desenvolvimento de habilidades de comunicação, escrita e apresentação por meio da defesa do PIE para uma banca avaliadora.

TÍTULO III

Condições para Realização

Art. 6 - O PIE deve ser desenvolvido em grupo, por estudantes que tenham cursado as disciplinas Construção de Algoritmos e Programação, Algoritmos e Estruturas de Dados 1 e Programação Orientada a Objetos.

TÍTULO IV

Oferta

Art. 7 - Serão lançados editais com periodicidade mínima anual contendo:

I - Texto caracterizando PIE e diretrizes gerais para o desenvolvimento do PIE;

II - Chamada e formato para a submissão de propostas PIE;

III - Datas para submissão, julgamento, divulgação e homologação dos projetos habilitados;

IV - Período para a inscrição das equipes nos projetos habilitados;

V - Divulgação dos projetos a serem desenvolvidos no período e as respectivas equipes.

Art. 8 - Os PIEs podem ser propostos por docentes, estudantes e empresas, sendo obrigatório que um docente da UFSCar atue como professor-orientador do projeto.

TÍTULO V

Caracterização do PIE

Art. 9 - Os PIEs devem, obrigatoriamente, empregar conhecimentos de 3 (três) ou mais disciplinas e se enquadrem como extensão, ou seja, possuírem potencial de atingir a comunidade externa ao Departamento de Computação.

Art. 10 - Os PIEs poderão contemplar práticas e/ou atividades como:

I - Projetos de pesquisa aplicada;

II - Elaboração de diagnósticos empresariais;

III - Projetos técnicos;

IV - Desenvolvimento de materiais didáticos e instrucionais;

V - Desenvolvimento de protótipos;

VI - Desenvolvimento de aplicativos e de produtos;

VII - Projetos de inovação tecnológica;

VIII - Outras modalidades reconhecidas como relevantes pela Coordenação de Curso.

Art. 11 - Para os projetos com potencial de inovação tecnológica, sugere-se que a equipe do projeto avalie a possibilidade junto Agência de Inovação da UFSCar de:

I - registrar o mesmo como registro de software no INPI ou divulgar como código-fonte aberto (repositórios), quando se tratar de desenvolvimento de software;

II - registrar como patente, quando de tratar de dispositivos de hardware.

TÍTULO VI

Visão Geral do Processo

Art. 12 - O processo de submissão, avaliação e acompanhamento de PIEs contém as seguintes atividades:

I - Submissão da proposta conforme cronograma previsto em edital específico;

II - Caracterização da proposta como integrador e extensão: haverá uma comissão definida em edital que avaliará se o projeto se caracteriza como integrador e extensão, emitindo um parecer no prazo estipulado no edital;

III - Cadastramento dos PIEs aprovados como atividade de extensão: para os projetos aprovados, o professor-orientador deve cadastrar o projeto submetido como atividade de extensão, dentro de programa de extensão específico, previamente cadastrado pelo coordenador do curso;

IV - Acompanhamento da execução do projeto: o professor-orientador deve acompanhar a execução do projeto e realizar avaliação individual e em grupo dos estudantes participantes;

V - Elaboração, sob a orientação do professor, de um relatório final, conforme modelo disponibilizado pela Coordenação de Curso;

VI - Apresentação dos resultados do PIE para banca examinadora, que poderá aprovar ou reprovar o resultado final obtido;

VII - Validação das horas dessa atividade pelos órgãos competentes.

CAPÍTULO I

Propostas

Art. 13 - A proposta para o PIE deve conter:

I - Capa

- a) Título;
- b) Áreas do Conhecimento/Disciplinas Contempladas;
- c) Sugestão de orientadores(as) ou indicar o orientador;
- d) Sugestão do tamanho da equipe necessária para o projeto (limite mínimo e máximo) ou indicar a equipe;

II - Contextualização;

III - Caracterização do problema;

IV - Justificativa;

V - Objetivos;

VI - Fundamentação Teórica (explicitando o vínculo com os conteúdos das disciplinas envolvidas);

VII - Metodologia;

VIII - Cronograma, incluindo:

- a) Atividades previstas, considerando a dedicação de 12 horas semanais por estudante;
- b) Previsão de entrega dos produtos do projeto;
- c) Datas de todas as reuniões presenciais e virtuais.

IX - Bibliografia.

CAPÍTULO II

Obrigações do Orientador

Art. 14 - As atividades relativas ao PIE serão supervisionadas pelo professor-orientador do Projeto Integrador que possui as seguintes obrigações:

I - Cadastrar o projeto como atividade de extensão junto à Pró-Reitoria de Extensão (ProEx);

II - Verificar o andamento das atividades de acordo com o cronograma submetido e aprovado;

III - Orientar os estudantes na condução das atividades;

IV - Registrar os encontros presenciais e virtuais.

CAPÍTULO III

Obrigações dos estudantes

Art. 15 - Aos estudantes cabe a realização das atividades do projeto, de acordo com o cronograma submetido e aprovado. Além disso, os estudantes devem:

I - Comparecer às reuniões presenciais e virtuais de acordo com o cronograma submetido e aprovado;

II - Dedicar pelo menos 12 horas semanais ao projeto.

TÍTULO VII

Avaliação

Art. 16 - A avaliação será composta de duas etapas:

I - A primeira etapa consiste em uma avaliação individual e contínua, e ficará a cargo do professor-orientador. Nesta etapa, serão considerados assiduidade e desempenho individual de cada estudante. Os estudantes reprovados nesta etapa serão desligados do projeto e não poderão ter direito à convalidação das horas da atividade;

II - A segunda etapa consiste em uma avaliação do projeto como um todo, que deve ser apresentado em forma textual (relatório final) e oral mediante uma banca examinadora. Nesta etapa, a banca examinadora irá avaliar o cumprimento da proposta aprovada, com atenção especial para o enfoque obrigatório de projeto integrador e

extensionista.

Art. 17 - Em termos de assiduidade, o aluno deve cumprir no mínimo 75% de frequência nas atividades do projeto.

Art. 18 - A banca será composta por um mínimo de três integrantes e um máximo de quatro, sendo pelo menos dois professores da UFSCar.

Art. 19 - O professor-orientador é membro natural da banca examinadora e irá presidir a sessão.

Art. 20 - A indicação de nomes de membro da banca, bem como a definição da data e reserva de sala é de responsabilidade do professor-orientador, respeitando o cronograma pré-estabelecido.

Art. 21 - Em caso de reprovação, o projeto poderá ser reapresentado, mediante solicitação por meio de formulário próprio, para a mesma banca examinadora.

Art. 22 - O estudante será reprovado automaticamente no Projeto Integrador quando ocorrer pelo menos um dos itens abaixo:

I - O trabalho não cumprir o objetivo proposto;

II - O trabalho for plágio;

III - O trabalho não for desenvolvido pelos estudantes;

IV - O trabalho estiver fora das normas técnicas exigidas pela Instituição;

V - O trabalho não for entregue no prazo estabelecido;

VI - Não for comprovada a presença de pelo menos 75% (setenta e cinco por cento) nas atividades do projeto.

Art. 23 - A ocorrência de qualquer dos itens anteriores deve ser comunicada pelo professor-orientador à Coordenação de Curso, que após avaliar a situação emitirá um parecer final.

Art. 24 - A conclusão com êxito de um PIE dará ao estudante direito a 210 (duzentas e dez) horas.

TÍTULO VIII

Obrigações da Coordenação de Curso

Art. 25 - Para garantir a oferta contínua de projetos em andamento, a coordenação de curso irá, a cada ano letivo, indicar dez docentes do Departamento de Computação que deverão submeter ao menos uma proposta de PIE naquele ano.

Art. 26 - A coordenação também será responsável por organizar e divulgar os editais de candidatura, aprovar as bancas de avaliação e validar as horas da atividade.

TÍTULO IX

Disposições Finais

Art. 27 - O presente Regulamento de Projetos Integradores Extensionistas entra em vigor a partir da data de sua aprovação pelo Conselho de Coordenação de Curso.

Art. 28 - Os casos omissos serão resolvidos pela Coordenação de Curso, cabendo ao Conselho do Curso analisar a necessidade de atualização deste Regulamento.